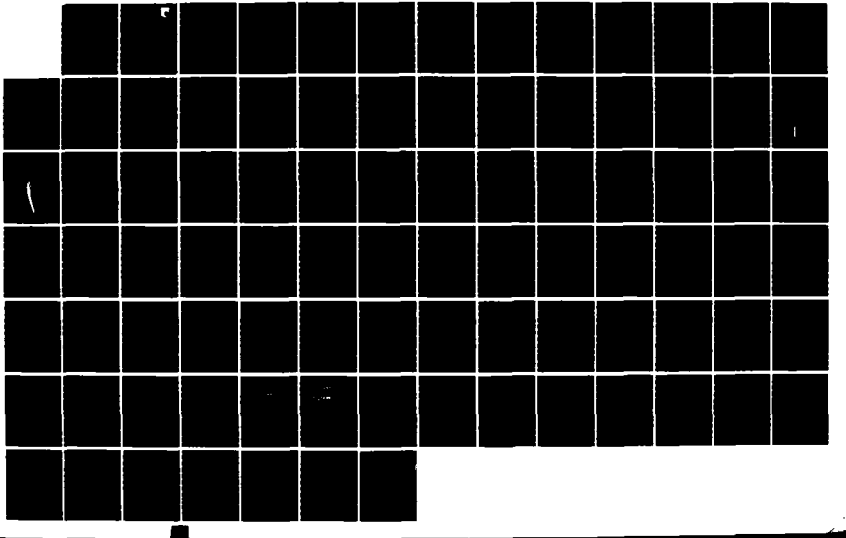
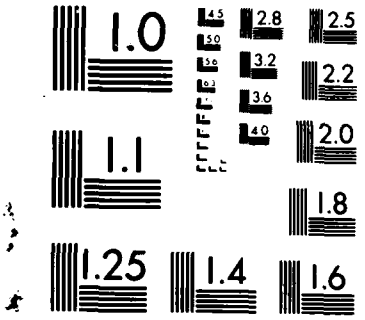
AD-A174 835 EXTENSION AND APPLICATION OF A PROCEDURE FOR DAMAGE ASSESSMENT OF AEROSPA..(U) DAYTON UNIV OH RESEARCH INST R F TAYLOR ET AL. SEP 86 UDR-TR-86-03 AFWAL-TR-86-3047 1/1

UNCLASSIFIED F33615-84-C-3406 F/G 1/3 NL





AFWAL-TR-86-3047

EXTENSION AND APPLICATION OF A PROCEDURE FOR DAMAGE ASSESSMENT OF AEROSPACE STRUCTURES

R. F. Taylor
R. A. Brockman
V. G. Dominic

University of Dayton
Research Institute
Dayton, Ohio 45469

September 1986

Final Report for Period May 1984 - November 1985

Approved for Public Release; Distribution Unlimited

FLIGHT DYNAMICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6553

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Gregory J. Czarnecki

GREGORY J. CZARNECKI
Project Engineer

AIVARS V. PETERSONS, Chief
Flight Vehicle Protection Branch
Vehicle Equipment Division

FOR THE COMMANDER

Rudi J. Berndt

RUDI J. BERNDT
Acting Chief
Vehicle Equipment Division

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFWAL/FIES, *W-PAFB, OH 45433 to help us maintain a current mailing list".*

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

AD-A174 035

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S): UDR-TR-86-03	5. MONITORING ORGANIZATION REPORT NUMBER(S): AFWAL-TR-86-3047
6a. NAME OF PERFORMING ORGANIZATION: University of Dayton Research Institute; 6b. OFFICE SYMBOL (If applicable):	7a. NAME OF MONITORING ORGANIZATION: AFWAL/FIES
6c. ADDRESS (City, State and ZIP Code): 300 College Park Dayton, Ohio 45469	7b. ADDRESS (City, State and ZIP Code): Wright-Patterson AFB, OH 45433
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: AFWAL; 8b. OFFICE SYMBOL (If applicable): FIES	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: F33615-84-C-3406
8c. ADDRESS (City, State and ZIP Code): Wright-Patterson AFB, Ohio 45433	10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.
62201F	2402	02	49

11. TITLE (Include Security Classification): EXTENSION AND APPLICATION OF A PROCEDURE FOR DAMAGE ASSESSMENT OF AEROSPACE STRUCTURES

12. PERSONAL AUTHOR(S): R. F. Taylor, R. A. Brockman, and V. G. Dominic

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
Final	FROM 5/15/84 TO 11/15/85	1986 September 30	88

16. SUPPLEMENTARY NOTATION

17 COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB. GR	structural analysis; vulnerability analysis; structural optimization; damage; stress analysis; deflection analysis; finite element analysis; automated design
01	03		

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This report documents the extension and application of a computer code for the Automated Design of Damage RESistant Structures (ADDRESS). The program is operable on the VAX-11/780 computer system under the VMS operating system. Extensions to the code over previous work includes the automated selection of damage locations via a sensitivity analysis, linking of design variables for more realistic damage tolerant member sizes, the introduction of standard material sizes in the output design, and the incorporation of a Sturm sequence check for determining frequencies in a given range. Applications are made to typical finite element models of lifting surfaces and to truss structures to validate the code for larger problems. By use of a setup procedure special purpose versions of the code can be made which include user specified upper bounds for key variables such as total number of nodes, elements, and degrees of freedom. Disk storage is the only limiting factor for larger problems, rather than program dimension specifications. All large-scale calculations are performed out of core.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☒ DTIC USERS ☐	UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE NUMBER (Include Area Code)	22c. OFFICE SYMBOL
Gregory J. Czarnecki	(513)255-6052	AFWAL/FIES

DD FORM 1473, 83 APR — EDITION OF 1 JAN 73 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

FOREWORD

This report documents work performed by the Aerospace Mechanics Division of the University of Dayton Research Institute (UDRI) for the Vehicle Equipment Division of the Flight Dynamics Laboratory, Air Force Wright Aeronautical Laboratories (AFWAL), Wright-Patterson Air Force Base, Ohio. The work was performed under Contract F33615-84-C-3406. Mr. Gregory J. Czarnecki was the AFWAL Project Engineer.

Included herein are the theory, numerical methods, and computation implementation of the procedures involved in development of an extended version of a computer code for the analysis and weight optimization of damage tolerant aerospace structures. User's information is provided for execution on the VAX-11/780 computer system.

The author ackowledges the helpful interaction with the AFWAL Project Engineer. Data for the unswept wing model, shown herein, was provided by the Air Force.

TABLE OF CONTENTS

SECTION		PAGE
1	INTRODUCTION	1
	1.1 Background	1
	1.2 General Approach	2
2	PROGRAM IMPROVEMENTS	3
	2.1 Increased Problem Size Capacity	3
	2.1.1 COMMON Block Reorganization	3
	2.1.2 Equation Solution Module	7
	2.1.3 Reanalysis and Sensitivity Analysis	10
	2.2 Material Dimension Requirements	11
	2.3 Selection of Damage Locations	12
	2.4 Sturm Sequence Based Vibration Analysis	14
3	REPRESENTATIVE APPLICATIONS	17
	3.1 Truss Structures	17
	3.2 Intermediate Complexity Wing	22
	3.3 Unswept Wing Structure	24
4	INPUT INSTRUCTIONS	29
5	PROGRAM OUTPUT	59
6	PLOTTING UTILITIES	63
	6.1 Geometry Plotting	64
	6.2 Contour Plotting	66
7	COMMAND PROCEDURES FOR VAX-11/780 OPERATION	73
	7.1 ADDRESS Installation	73
	7.2 ADDRESS Setup and Compilation	74
	7.3 ADDRESS Execution	77
8	SUMMARY AND CONCLUSIONS	79
REFERENCES		80

LIST OF ILLUSTRATIONS

FIGURE		PAGE
1	Stiffness Matrix Processing	9
2	Ten Bar Truss Geometry	18
3	Large-Scale Truss Geometry	20
4	Deformed Large-Scale Truss	21
5	Intermediate Complexity Wing	23
6	Unswept Wing Geometry	25
7	Undeformed Wing Upper Surface Plot	67
8	Deformed Wing Upper Surface Plot	68
9	Deformation Contour Plot for Upper Wing Surface	72

LIST OF TABLES

TABLE		PAGE
1	Summary of Adjustable Storage Parameters for ADDRESS	5
2	Member Sizes for Optimum Ten Bar Truss	19
3	ADDRESS Runs for Unswept Wing Structure	26
4	Sturm Frequency Checks for Unswept Wing Runs	28

SECTION 1
INTRODUCTION

Under a previous Air Force contract, research was conducted on the development of computerized methods for the optimization of damage resistant structures [1-2]. The computer program, ADDRESS (Automated Design of Damage Resistant Structures), was demonstrated to be a useful tool for the analysis and resizing of structures with ballistic damage. The effort documented herein was conducted in order to increase the efficiency and applicability of the code as discussed below. The improved source code is documented, demonstrated on several example problems, and is currently operational on VAX-11/780 computers with VMS operating systems. Reference 3 gives details of the optimization aspects of the program, and Reference 4 gives a development of the reanalysis method used in ADDRESS.

1.1 BACKGROUND

Military aircraft requirements of high performance and reduced structural weight are many times conflicting. This problem is particularly acute when the structure is subject to battle damage. A structure which is of some minimum weight for a given set of loads may be entirely inadequate when structural members are weakened or removed due to damage.

Under the original ADDRESS code development, it was shown that structural member sizes which give mean minimum weight can be determined so that all stress and deflection requirements in both the undamaged and damaged configurations can be satisfied. However, the application of optimization techniques to damaged structures raises a number of questions which have been subsequently addressed in this current work:

a. How should the structures be damaged?
b. What are the most critical damage locations?
c. How can localized resizing effects near damage locations be removed?
d. How can large sized problems be handled?

e. How can standard member sizes be determined for the structure?

1.2 GENERAL APPROACH

We rewrote the code, taking into account the benefits to be derived by using a computer with a virtual memory such as the VAX-11/780. This removed many of the problem size limitations which were inherent in the CYBER version of ADDRESS.

Since the VAX is slower in performing arithmetic calculations than the CYBER, care was taken in developing a faster equation solution technique. This approach has worked well as the results have been that the code now operates faster on the VAX than it did on the CYBER for many problems.

To solve the problems raised in the previous section we added more user inputs which would allow the questions to be answered on a problem-by-problem basis. Design variable linking, for example, can greatly reduce the number of variables in an optimization problem, and hence insight can be gained into determining critical damage locations by dealing with fewer independent variables. Variable linking is also used to eliminate the local redesign effects of damage. Handbook tables are included in the code to handle the standard member size problem.

SECTION 2
PROGRAM IMPROVEMENTS

This section describes the program improvements that were made to the ADDRESS code for operation on the VAX-11/780. Basic improvements to the operating efficiency and solution procedures are described in the first subsection. Material dimension requirements are then discussed, followed by considerations for the selection of the damage location. In the concluding subsection, frequency calculations based on Sturm sequence checks are described.

2.1 INCREASED PROBLEM SIZE CAPACITY

The original ADDRESS code had as a design objective the analysis and optimization of structural models containing several hundred elements, and a similar number of degrees of freedom. As such, the problem data were maintained in main memory, and a number of limitations were placed upon the maximum problem size. These limitations have been removed in the latest version of ADDRESS. The remainder of this subsection describes several modifications made to the code to remove restrictions on maximum problem size.

2.1.1 COMMON Block Reorganization

Virtually all of the major data storage in ADDRESS resides in COMMON blocks with fixed dimensions. For this reason, many of the inherent problem size limits are implicit in the declaration statements for these COMMON blocks.

The ADDRESS COMMONs have been extracted from the source code and placed in a single data file for easy revision. All array dimensions which impose problem size limitations are replaced in this data file by keywords having the form "$name", where the character string "name" describes the limit in question. For instance, "$maxel" appears wherever the corresponding

dimension must equal or exceed the number of finite elements in a model. A combination of FORTRAN code and DCL (Digital Command Language) command files process the COMMON data file in symbolic form to generate legitimate FORTRAN COMMON declarations, with each COMMON block contained in a separate file. The ADDRESS source code contains references to these files, in the form of INCLUDE statements.

With this organization of the COMMON data, a version of ADDRESS suitable for use with any size problem can be assembled, compiled, and linked within minutes. To set up a new version of ADDRESS with specified storage limits, a single procedure, SETUP, must be executed. This procedure is invoked by first setting the default directory to the ADDRESS installation directory, and then invoking the procedure. For example, if ADDRESS is installed on username USER under a subdirectory called ADDRESS, the following commands must be used:

```
$ Set Default [USER.ADDRESS]
$ @SETUP
```

The setup procedure will prompt for the problem size limits to be established; after this, the process is automatic. The following steps, which are transparent to the user, will be performed:

(a) INCLUDE files required for compiling ADDRESS are written;
(b) a DCL command procedure for compiling ADDRESS is created;
(c) a background task is entered to compile the program;
(d) a new executable version of ADDRESS is saved;
(e) the user is notified that the code is ready to use;
(f) the INCLUDE files and compilation procedure are deleted; and
(g) a log file summarizing the program limits is written.

Adjustable parameters which control the problem size capacity are summarized in Table 1. Most of the parameters are self-explanatory, in that they refer to an allowable number of nodes, elements, loading conditions, etc. The values for the

Table 1

Summary of Adjustable Storage Parameters for ADDRESS

Parameter	Description
$dkbuf	Length of buffer area for stiffness matrix changes
$maxbc	Maximum number of homogeneous constraints
$maxbl	Maximum number of matrix partitions
$maxdc	Maximum number of damage cases
$maxde	Maximum number of elements per damage case
$maxdf	Maximum number of degrees of freedom
$maxel	Maximum number of finite elements
$maxlc	Maximum number of load case (or modes)
$maxlm	Maximum number of lumped masses
$maxnd	Maximum number of node points
$maxur	Maximum number of user-level DAF records
$maxwk	Length of buffer for matrix storage

parameters $dkbuf, $maxbc, $maxbl, $maxur, and $maxwk control the amount of main memory available to ADDRESS for assembling and solving equations. They are briefly discussed below. In general, the default values for these memory related parameters are suitable for moderately-sized problems of several hundred elements, and they are set by the user as described and illustated in Section 7.

For $debuf, buffer length for stiffness matrix changes, the default is automatically set to 2000 without user intervention. This value will be adequate for nearly all problems unless very complex damage cases are involved, and then it must be increased. Dependence on data is complex, and the value can't be computed directly.

The $maxbc, maximum number of homogeneous constraints, is equal to the number of constrained degrees of freedom. The input instructions (data block G) in Section 4 describe the process for establishing the boundary conditions.

Parameters $maxbl and $maxwk determine the amount of storage available to ADDRESS for storing stiffness matrices for the system. These values must be larger than the matrix population as printed in the MATRIX STATISICS section of the output. Other than this, the choice depends on the system configuration, working set size, and other site parameters. In general, small $maxbl and large $maxwk uses more memory and less FORTRAN I/O, but more internal memory is then required. It is thus efficient if the working set is large. On the other hand, large $maxbl and small $maxwk uses very little memory, little paging, but this section requires more FORTRAN I/O which is often less efficient.

The $maxur parameter should be set to the default unless the $maxbl parameter is being changed. Both these parameters should be changed by roughly the same amount.

2.1.2 Equation Solution Module

Much of the primary data storage in any finite element code is determined by the methods and data structures used in forming and solving the final system of equations. In optimization and reanalysis, modifications of the stiffness coefficients must be considered, and storage demands become more serious still.

The equation solving module in ADDRESS has been completely rewritten to permit large problems to be solved efficiently. The following processes have been replaced:

- assembly of system stiffness and mass matrices;
- application of boundary conditions;
- matrix factorization;
- forward and backward substitutions; and
- recovery of displacement results.

In addition, some operations associated solely with reanalysis have been replaced. These are discussed separately in Paragraph 2.1.3.

The revised assembly and solution procedures are oriented toward out-of-core and virtual memory operation. In a static analysis, for example, it is necessary to solve the system

$$\mathbf{Kx} = \mathbf{F}$$

The solution is obtained in four steps as follows:

Factorization:	$\mathbf{K} = \mathbf{L}\mathbf{D}\mathbf{L}^T$
Forward Substitution:	$\mathbf{L}\mathbf{z} = \mathbf{F}$
Scaling:	$\mathbf{D}\mathbf{y} = \mathbf{z}$
Back Substitution:	$\mathbf{L}^T\mathbf{y} = \mathbf{x}$

Matrix L is unit lower triangular; that is:

$$L_{ij} = 1 \qquad \text{for } i = j$$
$$L_{ij} = 0 \qquad \text{for } i < j$$

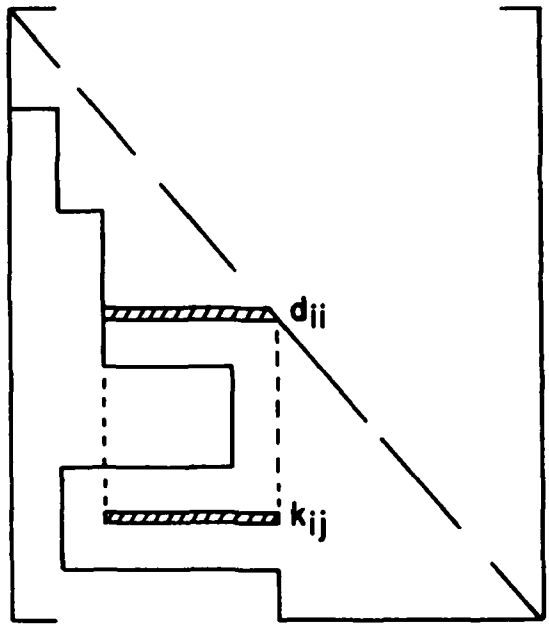
and **D** is diagonal. The factorization preserves the sparsity pattern of the original stiffness matrix; therefore matrix **L** may overwrite the subdiagonal portion of **K**, and **D** is stored on the diagonals of **K**.

During the factorization of **K** into the product $\mathbf{LDL}^T$, values of matrix elements corresponding to each column containing nonzero elements of a particular row are needed each time a new row is processed (Figure 1). Since the stiffness matrix resides on disk (either in virtual memory or because of explicit file operations), the basic algorithm for factoring **K** requires an excessive amount of disk I/O. The factorization step in ADDRESS has been reordered to simulate the sequence of operations needed to minimize I/O activity on a small, fixed-memory machine.

Depending on the work space available to the program, the matrix is partitioned by rows and stored as a direct access file. If the work space is large, the number of partitions is small, and disk I/O is associated primarily with paging by the virtual memory system. When a small work space is declared, this I/O is performed in FORTRAN by ADDRESS. The reordering of the factorization step serves to minimize either type of I/O activity.

The forward substitution pass is critical in terms of I/O activity as well. In the basic algorithm, each unknown is evaluated using a single matrix row: this means that each coefficient in a row must be accessed before proceeding to the next. Near the end of the forward substitution, the entire matrix must be read from disk to process each equation! In ADDRESS, the basic forward substitution algorithm has been revised to accumulate the sums for all rows in parallel, so that the matrix must be read only once for the entire procedure. As with the factorization, both FORTRAN I/O and virtual memory activity are minimized with this modification.

The assembly of the system stiffness matrix presents some problems when the assembled matrix is stored on disk, either



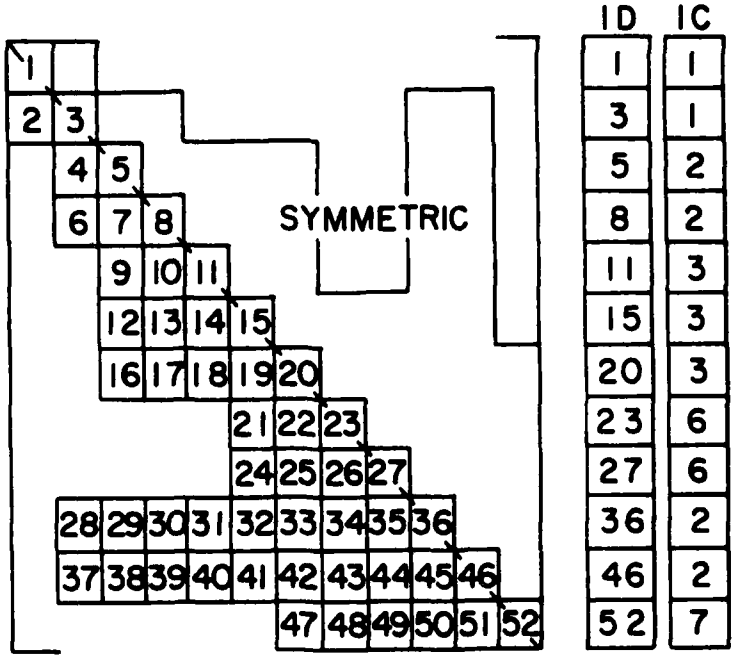


Figure 1. Stiffness Matrix Processing.

as a random access file or in virtual memory. Since many elements in a large model may contain entries which belong in more than one partition (or page) of the matrix, the complete matrix may be read literally hundreds of times during a single assembly pass. To minimize this effect, element matrices are computed and written to a temporary sequential file before the assembly begins. The assembly is then performed by partitions of the system matrix, so that each partition is written only once, and never reread. The matrix assembly module uses as much space as possible to minimize the number of passes through the element files (usually two partitions can be assembled at a time).

The method of enforcing boundary conditions in ADDRESS has been modified to reduce storage requirements and I/O. Additional pointer information is now maintained which locates constrained degrees of freedom, which are skipped during the matrix assembly process, rather than being compressed out afterward. Procedures for recovering displacement results and reactions have been modified as well for compatibility with the revised method.

The revised assembly and solution module has exhibited no problems with excessive I/O activity in tests performed thus far. Another benefit is the reduction in CPU time, which is typically a factor of three or four from the original code. The efficiency improvement is due partially to reduced I/O; however, smaller problems for which I/O is minimal also show a substantial increase in efficiency.

2.1.3 Reanalysis and Sensitivity Analysis

In performing reanalysis for structural damage conditions, a typical iteration consists of updating the solution using the recursion formula

$$\mathbf{K}\mathbf{x}^{(i+1)} = \mathbf{F} - d\mathbf{K}\mathbf{x}^{(i)}$$

The first major step in each iteration involves forming the product **dKx**; here **dK** represents the modifications to the system stiffness matrix for a given damage condition, and **x** is the latest estimate of the displacement solution. A similar operation is required in sensitivity analysis (see subsection 2.3), in which the sensitivities **x'** represent derivatives of the response with respect to a damage parameter:

$$\mathbf{Kx' = -K'x}$$

In each case, the factors of **K** are already known, and the solution is obtained by performing simple forward and back substitutions.

In practice, the changes in stiffness **dK** and the derivatives **K'** are nonzero only for selected elements of the model, but the assembled matrices **dK** and **K'** may require substantial storage. The procedure used originally to assemble these matrices has been replaced to remove all restrictions on their size. Rather than assembling the modified stiffness directly, we note that it is always the product **dKx** (or **K'x**) which is required. This product is formed element by element, and then assembled in a single array whose length equals the number of unconstrained degrees of freedom in the model. Since the element matrices which must be modified are now saved on temporary files for assembly, the recalculation of these matrices has been eliminated as well.

2.2 MATERIAL DIMENSION REQUIREMENTS

A recurring problem with computed optimum designs is that the final member sizes do not correspond to standard material gages. An option has been added to the optimization branch of ADDRESS to compute near-optimum member sizes based on specified standards for material stock dimensions.

During the optimization process, member sizes may assume arbitrary values between their specified minimum and maximum values. However, when output is requested in terms of standard

material gage sizes, an additional analysis pass is made to determine acceptable dimensions for each member and to analyze the resulting final design.

The determination of standard material dimensions is performed in subroutine STDSIZ within ADDRESS, where acceptable gage sizes are tabulated in the form of DATA statements. The standard gages used in ADDRESS are based upon the following sets of dimension standards:

- Metal sheet: Manufacturer's Standard Gauge
- Steel bar: U. S. Steel Wire Gauge
- Bar (other): American Wire Gauge, Non-ferrous

Standard dimensions for each of these systems are tabulated in ADDRESS in subroutine STDSIZ, in the form of DATA statements. These three-dimension standards encompass stock sizes through 0.2391 in., 0.4900 in., and 0.651354 in., respectively. For larger sheet thicknesses or bar diameters, multiples of 1/16 in. are used.

For bar elements, the material modulus and density are tested to determine if the element is steel, so that appropriate standard gauges can be used. Presently, these tests are expressed in English (lb-in-sec) units only.

2.3 SELECTION OF DAMAGE LOCATIONS

To assist in the selection of damage locations, a procedure for sensitivity analysis has been added to the ADDRESS program. In what follows, we first describe the problem of sensitivity analysis in general terms, and then discuss the procedure implemented in ADDRESS.

Consider the static response problem, which for a finite element model of a structure takes the form

$$\mathbf{Kx} = \mathbf{F}$$

In general, the stiffness coefficients are functions of a set of design variables p ; in ADDRESS, these variables correspond to member areas or thicknesses. The sensitivity of the response to

any one of these design variables is measured by the response derivatives. For example, the element stiffness matrix for a bar is a linear function of its cross-sectional area, A. Thus

$$K_{Bar} = P\ (\frac{E}{\ell})\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

where **P**=A, **E** is Young's modulus, and ℓ is the length of the bar.

If we denote the derivative with respect to one of these parameters by ( )', then:

$$(\mathbf{Kx})' = \mathbf{F}'$$

In many cases, the applied forces do not depend upon the design variable in question, and F'=0. In general, we can solve for the sensitivities x' using:

$$\mathbf{Kx}' = \mathbf{F}' - \mathbf{K}'\mathbf{x}$$

To perform this solution, assuming the original problem Kx = F has been solved, we must (a) compute the derivative terms K'x; and (b) perform forward and back substitutions using the factors of K.

For the elements in ADDRESS, the stiffness matrix for an element depends linearly upon the corresponding design variable, so that:

$$\mathbf{K}' = \mathbf{K}/t \quad \text{for 2-D elements}$$
$$\mathbf{K}' = \mathbf{K}/A \quad \text{for bar elements}$$

The terms K'x therefore are readily formed at the element level, and assembled in the same way as element applied forces.

The computation of sensitivities with respect to individual design variables of the model is simple and inexpensive. However, one can imagine examining literally hundreds of cases of this form to determine critical locations for damage sensitivity. Instead, we define a damage parameter q, which in effect couples many of these cases together for use in sensitivity analysis. A unit change in the value of q is defined (through user input) as corresponding to a thickness or area reduction of f(n) in element

e(n), for n=1,2,...,N. The sensitivities then represent derivatives with respect to damage conditions which involve equal or proportional changes in a series of elements. Once the displacement sensitivities have been computed, the sensitivities of the element stresses can be computed directly.

Typically, the collection of elements corresponding to a single sensitivity parameter represent a localized region of the model for which the sensitivity to damage is to be examined. No limits are placed upon which elements, or how many elements, are linked in a particular sensitivity case, however. The definition of the generalized sensitivity parameter q is performed in much the same way as the specification of actual damage conditions (see Section 4).

2.4 STURM SEQUENCE BASED VIBRATION ANALYSIS

During the design and analysis of structures, it is many times important to know the free vibration frequencies and mode shapes. The location of natural frequencies is important since resonance problems may occur which can lead to excessive amplitudes of motion and hence large dynamic stresses.

The vibration analysis requires that the matrix eigenvalue problem

$$\mathbf{K}x_i = {}_i\mathbf{M}x_i \qquad i = 1,\ldots,m\leq n \qquad (1)$$

be solved where λ_i and x_i are the natural frequency squared and mode shape for the I-th mode, respectively. The assembled structural stiffness matrix, **K**, and mass matrix, **M**, are of order n, symmetric, nonnegative definite, and sparsely populated for the finite elements used by the ADDRESS code.

Many eigenvalues, including the ones used previously in ADDRESS, determine the eigenvalues in order starting with the smallest until some specified number m are computed. Thus,

$$\lambda_1 < \lambda_2 < \ldots < \lambda_{m-1} < \lambda_m$$

The Sturm sequence method does not find them from smallest to largest, but instead finds all of them within a specified range. So,

$$\lambda_\ell < \lambda_j < \lambda_{j+1} < \ldots < \lambda_{j+r} < \lambda_u$$

where r eigenvalues lie in the interval from λ_ℓ to λ_u.

As a practical example of this use of property, it may be important to determine if the structure has any natural frequencies in the range 50 to 100 Hz since a motor also has an operating frequency in this range. The ADDRESS code will determine the frequencies in any given range for the unmodified structure as well as the damaged configurations if needed.

According to the theory of Sturm sequences [5], for any given trial value of λ, the factorization

$$\mathbf{A} \equiv \mathbf{K} - \lambda\mathbf{M} = \mathbf{L}\ \mathbf{D}\ \mathbf{L}^T \qquad (2)$$

is required to establish the frequency count. The key result is that the number of negative elements in **D** is equal to the number of eigenvalues smaller than λ. Thus if **D** has p negative terms at $\lambda = \lambda_\ell$ and q negative terms at $\lambda = \lambda_u$, there are exactly q - p eigenvalues in the range between λ_ℓ and λ_u. The ADDRESS code contains an iteration technique which solves for all eigenvalues in a given range up to a specified tolerance by subdividing the interval into smaller segments and isolating the values.

Eigenvectors are determined by inverse iteration [6]. This procedure is developed as follows. Let μ be a scalar and subtract $\mu\mathbf{M}$ from both sides of Equation 1 to obtain

$$(\mathbf{K}-\mu\mathbf{M})\mathbf{x}_i = (\lambda_i-\mu)\mathbf{M}\mathbf{x}_i$$

or

$$(\lambda_i-\mu)^{-1}\ \mathbf{x}_i = (\mathbf{K}-\mu\mathbf{M})^{-1}\ \mathbf{M}\ \mathbf{x}_i$$

This is written as

$$\mathbf{B}x_i = p_i x_i \tag{3}$$

where

$$\mathbf{B} = \mathbf{A}^{-1}\,\mathbf{M}$$
$$\mathbf{A} = \mathbf{K} - \mu\mathbf{M}$$
$$p_i = (\lambda_i - \mu)$$

In the case that $\lambda \approx \lambda_i$, Equation 3 is treated as an iteration formula

$$\bar{x}^{(k+1)} = \mathbf{B}\, x^{(k)} \tag{4}$$

for a selected value of μ. If μ is close to λ, then Equation 4 will converge very rapidly (two or three iterations) to the eigenvector x_i

The **B** matrix is never formed. Rather, the equivalent system of equations

$$\mathbf{A}\,\bar{x}^{(k+1)} = \mathbf{M}\, x^{(k)} \tag{5}$$

is solved using the equation solver of Section 2.1.2. After $\bar{x}^{(k+1)}$ is obtained it is normalized and becomes the vector on the right hand side of Equation 5 during the next cycle.

SECTION 3
REPRESENTATIVE APPLICATIONS

Several structures are selected which show the analysis and optimization capability of the VAX version of ADDRESS. The ten bar truss and intermediate complexity wing are the same structures as presented in the Reference 1 development. A larger scale truss structure is presented to validate that the program is operational on problems with a large number of elements, modes, and degrees of freedom. An unswept wing structure, based on data received from the Flight Dynamics Laboratory, is also presented. This last structure is similar to the one that may be constructed and tested.

3.1 TRUSS STRUCTURES

The ten bar truss (Figure 2) described in the original ADDRESS documentation [1-2] has been optimized using the revised ADDRESS code. The primary reason for performing this example was to validate corrections made to the stress constraint algorithm. In the original ADDRESS report, an optimum design for two loading conditions was obtained, with a total weight of 5455 pounds. Subsequent modifications to the resizing strategy produced a slightly improved design weighing 5336 pounds. The latest version of ADDRESS produces a design weighing 5091 pounds, an improvement of nearly five percent. The final member sizes are summarized in Table 2.

To test the ability of ADDRESS to analyze larger problems, a large-scale truss containing 3100 elements has been considered. The geometry is shown in Figure 3 and resembles that of a bridge. The problem considered consists of a static analysis, with a downward force applied at center span; since both the geometry and applied force are symmetric, only half the truss is modeled. All members are steel, with a modulus of 30 Mpsi with a total force of 10,000 pounds applied. The deformed shape of the bridge is shown in Figure 4. The total number of degrees of freedom is

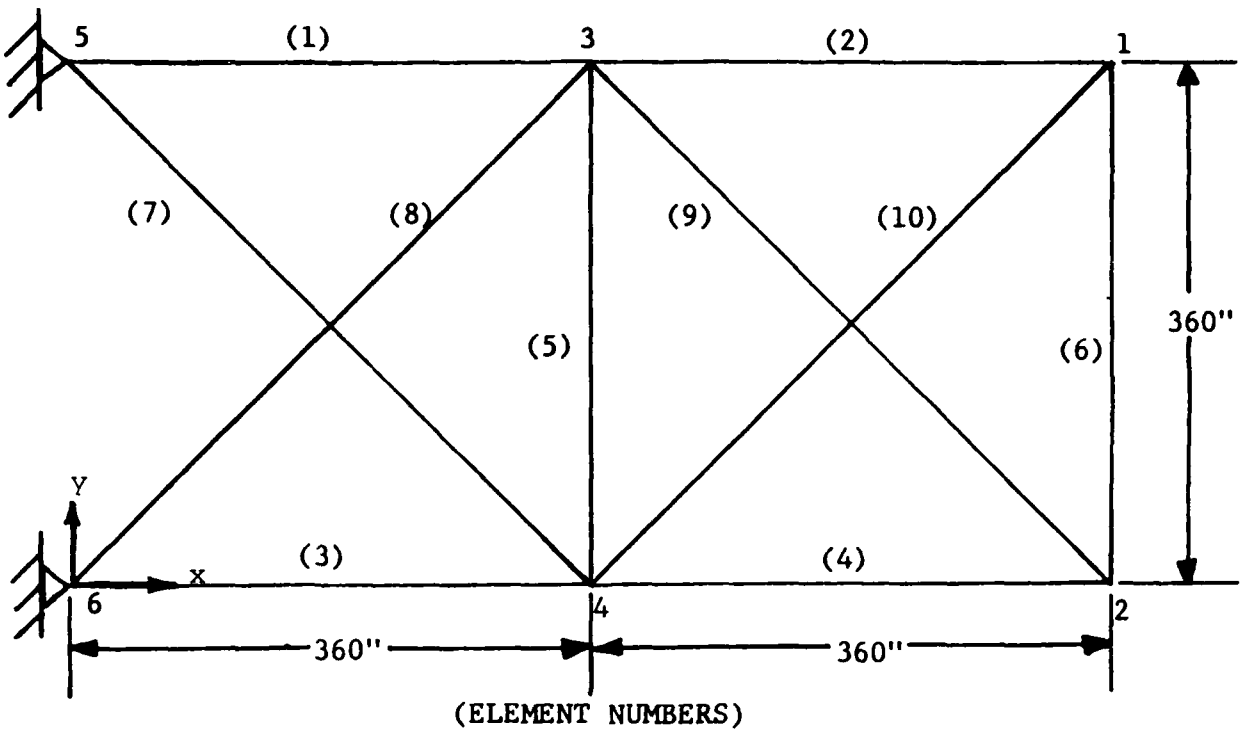


Figure 2. Ten Bar Truss Geometry.

TABLE 2

MEMBER SIZES FOR OPTIMUM TEN BAR TRUSS

Element Number	Cross-Sectional Area (in^2)
1	30.595
2	0.010
3	24.296
4	14.607
5	0.010
6	0.557
7	9.114
8	20.672
9	20.657
10	0.010

Total Weight = 5091.33 lb.

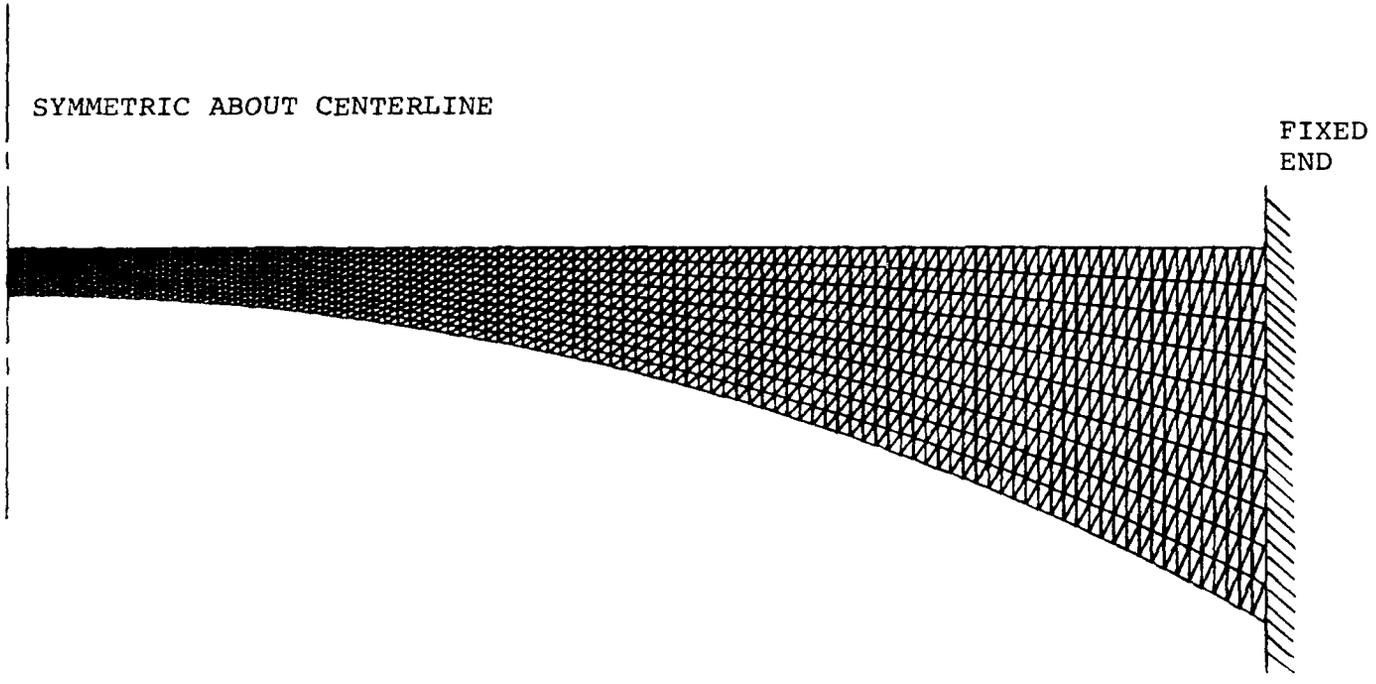


Figure 3. Large-Scale Truss Geometry.

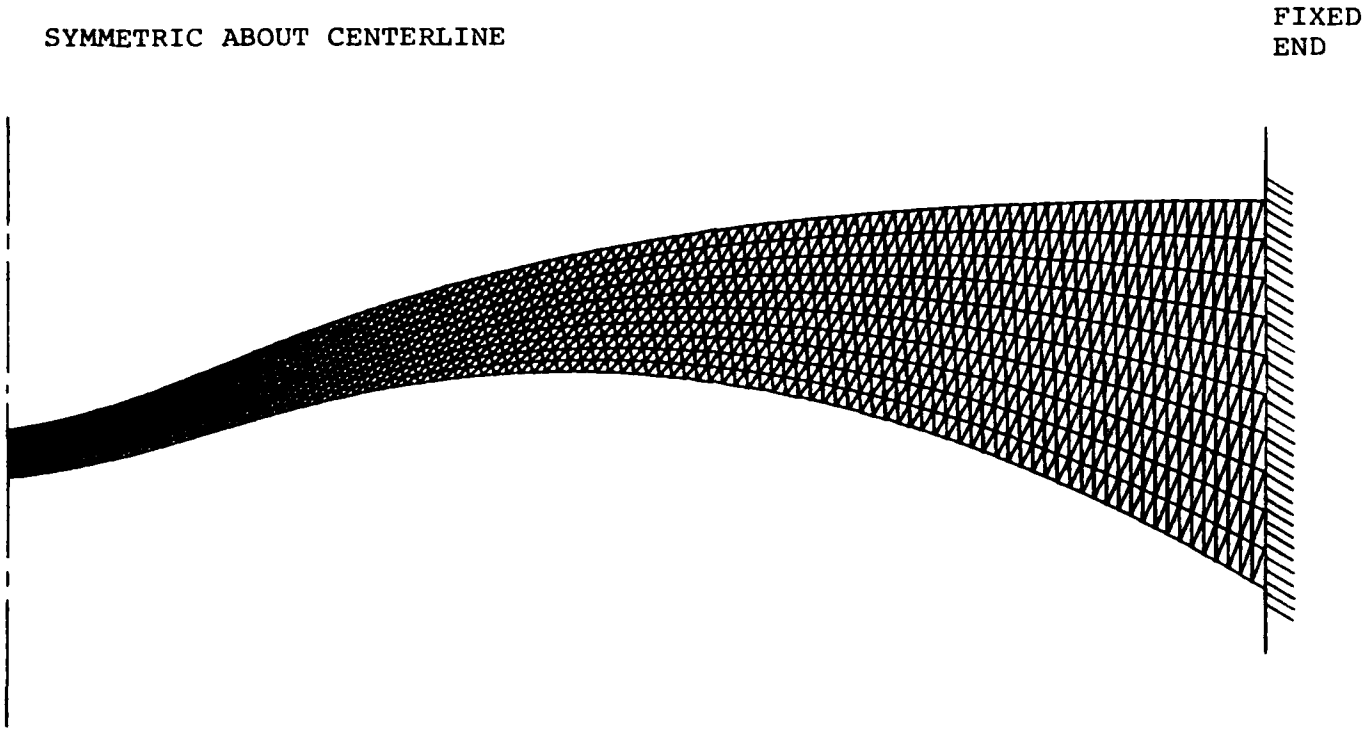


Figure 4. Deformed Large-Scale Truss.

still relatively small (2,189), but sufficient to force multiple partitions in the equation solution phase. The static analysis and stress recovery required 224 CPU seconds on the VAX-11/780.

3.2 INTERMEDIATE COMPLEXITY WING

The intermediate complexity wing structure considered in the original ADDRESS document [1], Section 5.3, has been analyzed with the VAX version of ADDRESS to provide performance benchmark data on the revised program. The model, shown in Figure 5, is comprised of 158 elements.

The ICW model has been analyzed on the VAX with two versions of ADDRESS. The first analysis was performed after minimal code modifications, made solely for compatibility with the VAX. The second analysis uses the final version of ADDRESS, with the newer assembly and solution modules. The overall solution time for a static analysis is reduced by a factor of three in the final code, with the matrix assembly and solution time a factor of six faster. The static analysis on the VAX, which requires 35.1 CPU seconds with the revised ADDRESS, compares well with the CYBER-74 analysis time of 20.4 CPU seconds using the original code.

Reanalysis and sensitivity analysis have been tested also for the ICW. For a single damage case, the total time for analysis and reanalysis on the VAX is 41.9 CPU seconds. That is, the reanalysis consumes about 6.8 seconds, less than 20% of the time required for a complete resolution. A sensitivity analysis requires substantially less time than reanalysis, since it is roughly equivalent to a single iteration of the reanalysis procedure. In a case involving the same number of elements as the damage reanalysis, the sensitivity analysis requires about 4.4 CPU seconds, or 12.5% of the time required for the original solution.

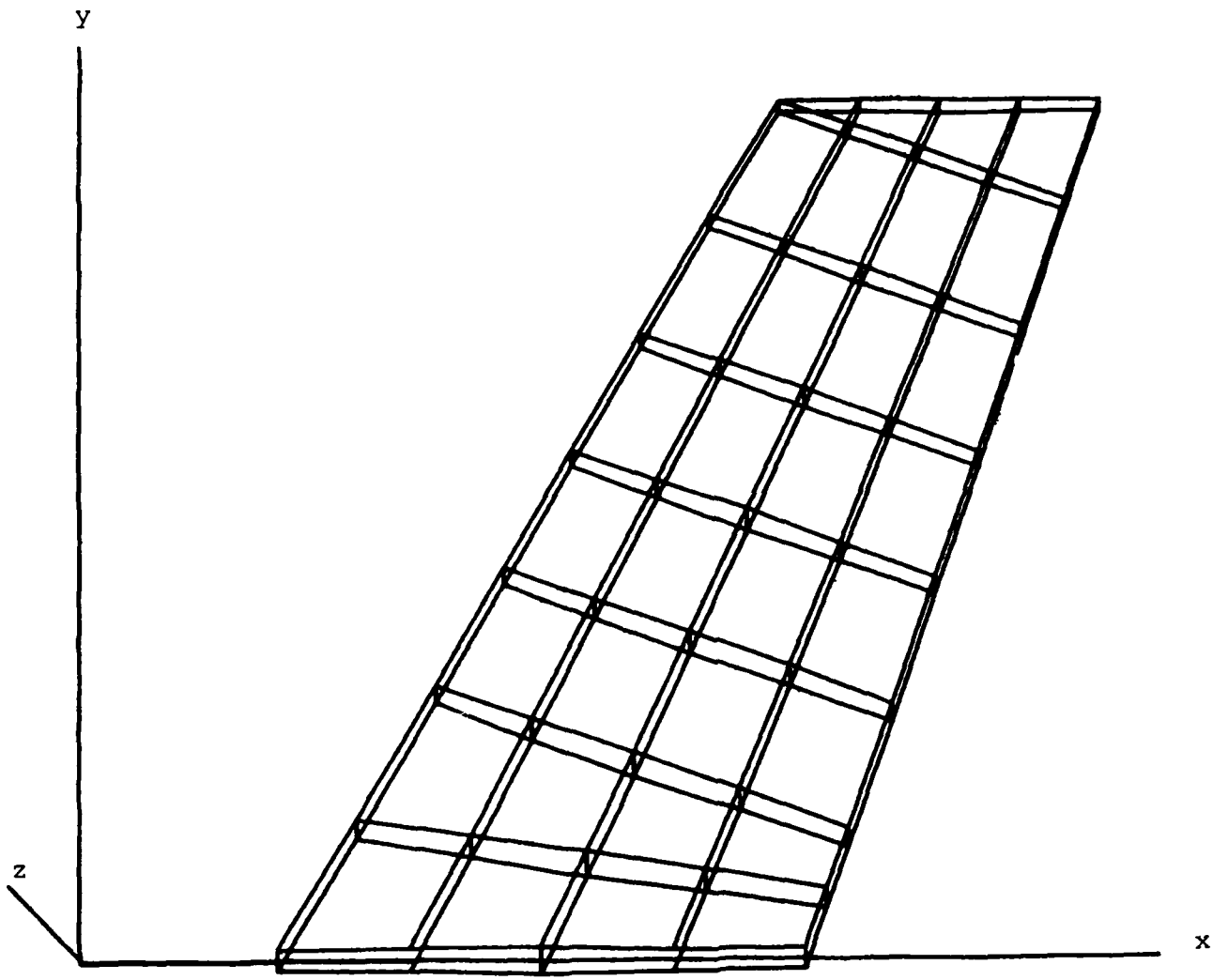


Figure 5. Intermediate Complexity Wing.

3.3 UNSWEPT WING STRUCTURE

Figure 6 shows the finite element model of an unswept wing structure used to test the optimization and damage assessment capabilities of the ADDRESS program. The modeling has the following features:

- All metal construction (Young's modulus 10^7 psi, weight density 0.1 lb/in^3)
- 96 nodes (6 fixed at root)
- 179 elements
- 264 unrestrained degrees of freedom

A total of nine ADDRESS runs were performed as shown in Table 3.

The analysis of Run 1 shows that the stresses are very low in the covers ranging from 109.7 psi (direct) at the tip to 1558 psi (shear) at the root. The stress critical elements are in the spars between Rib 7 and the root where they are as high as 24.71 KSI (shear).

Run 2 shows similar analysis results for the stresses in the covers. Shear stresses near the root have peak values of 36.53 KSI (shear). Deflections of the structure for Run 2 are 3.097 inches at the tip. This compares with 1.001 inches for Run 1.

Stress optimization is performed in Run 3. The cover elements are resized from an initial value of 0.07 inches in thickness down to 0.001 inches at the tip to 0.03 inches between Rib 7 and the root. Stresses are very close to their allowables which are set at 25 KSI. Spar elements near the root increase from their initial values of 0.02 inches to 0.05 inches. No elements are critical for this design. The structural weight for the stress design is 17.88 lb. which compares to 75.26 lb. for the initial design.

Stress optimization results for Run 4 are similar in overall trends to those of Run 3. However, the total optimum weight is 26.54 lb. Most of the additional weight is in the

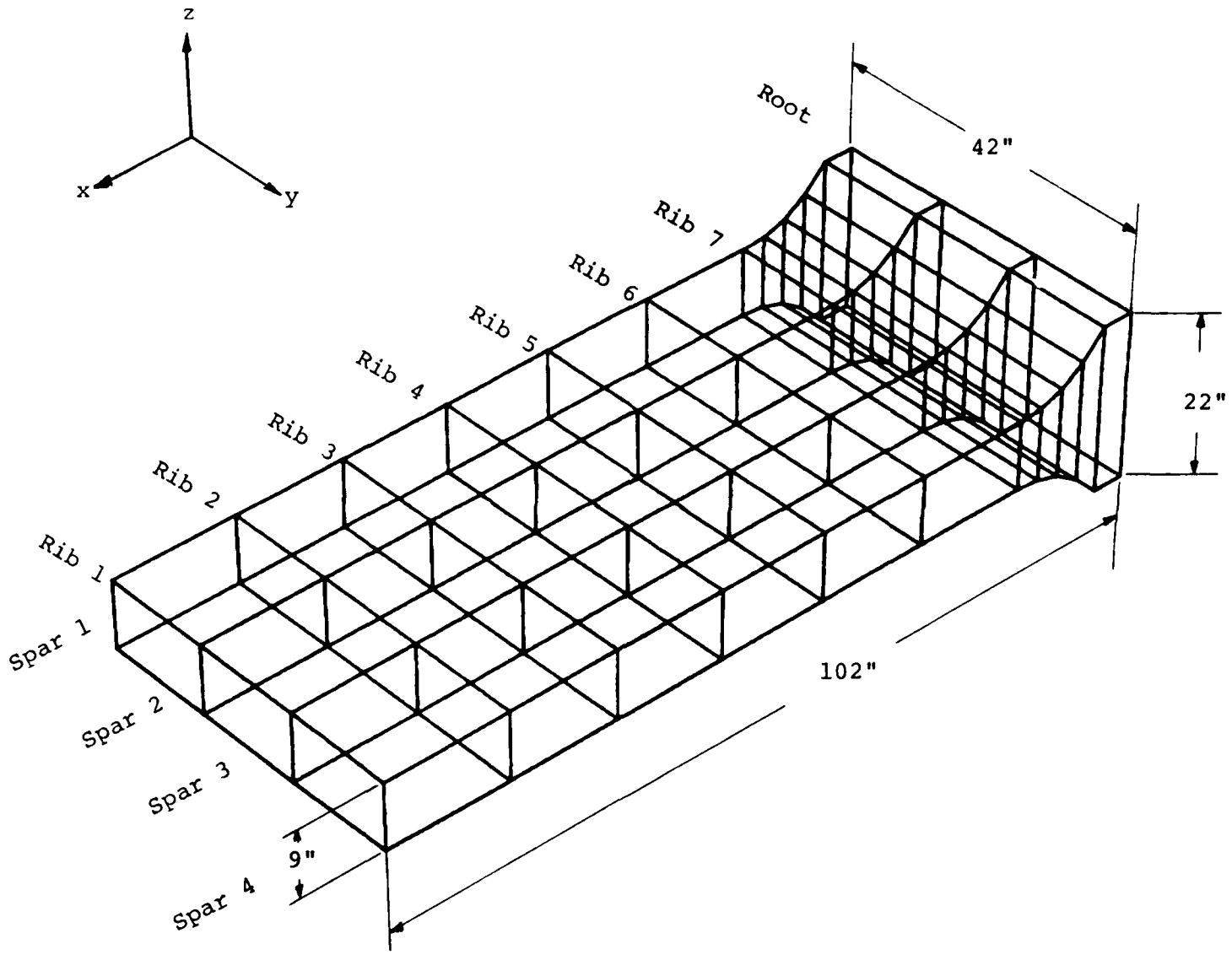


Figure 6. Unswept Wing Geometry.

TABLE 3

ADDRESS RUNS FOR UNSWEPT WING STRUCTURE

Run Number	Description
1	Static analysis with 100 lb. at each node in Z direction
2	Static analysis with 300 lb. along Ribs 1 and 2
3	Same as Run 1 with stress optimization
4	Same as Run 2 with stress optimization
5	Same as Run 2 with stress optimization and damage
6	Frequency checks for Run 1 initial design
7	Frequency checks for Run 3 final design
8	Frequency checks for Run 4 final design
9	Frequency checks for Run 5 final design

covers near the tip. Spars are also increased in thickness near the tip.

In Run 5, damage is included for all elements between the third and fourth ribs at the second and third spars. It is assumed that 50% of the material is removed from the covers, ribs, and spar in this zone. The weight increases to 35.87 lb., compared to the optimum, but not damage tolerant, weight of 26.54 lb. Cover and rib elements adjacent to the damage are increased by about 50% in thickness. Elements near the tip and root are not greatly affected by the damage with resizing increases of 10%.

Rows 6 through 9 involve the use of the Sturm sequence frequency checking. Table 4 summarizes the results. It can be seen that optimization generally shifts the frequencies to lower values. This is especially true of the Run 7 design which is optimum for the distributed load. The introduction of damage tolerance in the Run 9 design moves one of the frequencies above 1000 Hz which was previously below this value. Most of the frequencies of interest for ballistic damage consideration would likely be below 100 Hz. Since this structure is quite light and stiff, there are only several frequencies in this range depending on the design under consideration.

TABLE 4

STURM FREQUENCY CHECKS FOR UNSWEPT WING RUNS

Check Frequency (Hz)	Number of Frequencies Less Than Check Frequency			
	Run 6 (Baseline)	Run 7 (Opt-Distributed)	Run 8 (Opt-Tip Load)	Run 9 (Opt-Damage)
1	0	0	0	0
10	0	0	0	0
100	2	4	2	2
1,000	60	58	51	50
10,000	228	234	234	234

SECTION 4
INPUT INSTRUCTIONS

In this section the data requirements of the ADDRESS program are discussed. The input consists of the following blocks of data:

A. General Input
B. Coordinate Date
C. General Element Data
D. Materials Data
E. Connectivity Data
F. Fiber Orientation Data (optional)
G. Boundary Conditions
H. Loads Data
I. Displacement Data (optional)
J. Damage Data (optional)
K. Lumped Mass Data (optional)

The following pages describe each of these data blocks in detail. Each data block consists of one or more card sets, and each card set consists of one or more cards. The notes inform the user as to the number of cards in each set as well as explanatory information.

Each data item is given an identification code of the form L.M.N. The "L" is a letter which refers to the above data blocks; "M" and "N" are integers which refer to the card set and the item within the card set, respectively. Thus, for example, the code "E.1.4" refers to the connectivity data block, card set1, data item 4. Variables marked with an asterisk (*) indicate that the user should refer to the Notes at the end of the input for the particular block for additional information.

Data block A is very important. All the program options are selected by specifying values on this record. The user should always refer to this card first before making data changes in order to assure that the proper program options have been selected.

INSTRUCTIONS

ADDRESS DATA BLOCK: A

TITLE: General Input

Inclusion Conditions: Required

Input Subroutine(s): INITIAL (Card Set 1)
INGNRL (Card Set 2-5)

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
A.1	A.1.1 (I5)	NSTR (1-5)	Number of problems to be solved (Card Set A.1 contains one card)
A.2	A.2.1 (A80)	TITLE (1-80)	Any alphanumeric description of the problem (Card Set A.2 contains one card)
A.3	A.3.1 (I5)	MM (1-5)	=2 if problem is two dimensional =3 if problem is three dimensional
	A.3.2 (I5)	LMTDSP (6-10)	=0 if no displacement constraint =1 if displacement constraint same at all nodes =2 if displacement constraint not the same at all nodes
	A.3.3 (I5)	LMTSTR (11-15)	=0 if no stress constraint =1 if stress constraint
	A.3.4 (I5)	NDMGCAS (16-20)	Number of damage cases
	A.3.5 (I5)	NLMPMSS (21-25)	Number of lumped masses
	A.3.6 (I5)	MAXDCCL* (26-30)	Number of iterations for optimization with displacement constraints
	A.3.7 (I5)	MAXECCL* (31-35)	Number of iterations for optimization with stress constraints
	A.3.8 (I5)	IAREAS* (36-40)	=0 if initial design variables of the elements are to be set to 1.0 =1 if initial design variables are input for each element

*See Notes

INSTRUCTIONS

ADDRESS DATA BLOCK: A

TITLE: General Input (continued)

Inclusion Conditions: Required

Input Subroutine(s): INITIAL (Card Set 1)
INGNRL (Card Set 2-5)

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
	A.3.9 (I5)	INDMIN* (41-45)	=0 if minimum allowable size is the same for all elements =1 if minimum allowable size is input for each element
	A.3.10 (I5)	KANALYZE* (46-50)	=0 if optimization is to be performed =1 if analysis only is to be performed
	A.3.11 (I5)	MAXSZE* (51-55)	=0 if no maximum size is to be input =1 if maximum size is to be input
	A.3.12 (I5)	LAYERD (56-60)	=0 if problem contains no layered composites =1 if problem contains layered composites
	A.3.13 (I5)	INDANG (61-65)	=0 if problem contains no layered composites or if zero degree fibers are defined per element with respect to the global coordinate system =1 if the zero degree fibers are defined per element with respect to the local coordinate system =2 if zero degree fibers are oriented the same for all elements with respect to the global coordinate system
	A.3.14 (I5)	MNLAYR* (66-70)	=0 if problem contains no layered composites or if the composite layup total thickness remains constant. =1 if the minimum total thickness of 0°, 90°, ±45° layers will be input for each element

*See Notes

INSTRUCTIONS

ADDRESS DATA BLOCK: A

TITLE: General Input (continued)

Inclusion Conditions: Required

Input Subroutine(s): INITIAL (Card Set 1)
INGNRL (Card Set 2-5)

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
	A.3.15 (I5)	IPP (71-75)	=0 if a postprocessor file not needed =1 if a postprocessor file, DPOST, will be created
	A.3.16 (I5)	IROUND (76-80)	=0 if standard sizes will not be used =1 if standard sizes will be used
			(Card Set A.3 contains one card)

INSTRUCTIONS

ADDRESS DATA BLOCK: A

TITLE: General Input (continued)

Inclusion Conditions: Required

Input Subroutine(s): INITIAL (Card Set 1)
INGNRL (Card Set 2-5)

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
A.4	A.4.1 (F10.3)	AEMNMM* (1-10)	Minimum allowable design variable (thickness or area)
	A.4.2 (F10.3)	DINCR* (11-20)	Parameter to determine the active set of displacement constraints
	A.4.3 (F10.3)	THKLAM* (21-30)	Minimum composite layer thickness
	A.4.4 (F10.3)	SPRDF* (31-40)	Shear panel reduction factor
			(Card Set A.4 contains one card)

*See Notes

INSTRUCTIONS

ADDRESS DATA BLOCK: A

TITLE: General Input (continued)

Inclusion Conditions: Required

Input Subroutine(s): INITIAL (Card Set 1)
INGNRL (Card Set 2-5)

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
A.5	A.5.1 (I5)	MODES (1-5)	Number of modes to be computed
	A.5.2 (I5)	NOI* (6-10)	Number of iterations to determine modes
	A.5.3 (F10.3)	TOLVEC* (11-20)	Tolerance on modes
	A.5.4 (I5)	NSTURM* (21-25)	=0 if no Sturm frequency checks are to be performed >0 if Sturm frequency checks are to be performed (Card Set A.5 contains one card)
A.6	A.6.1 (F10.3)	FBDS(I), I=1,NSTURM (1-10,11-20,etc.)	Frequencies (Hz) for Sturm frequency checks. Include card set only if NSTURM is not zero. (Card Set A.6 contains 8 values of FBDS per card)

*See Notes

NOTES ON DATA BLOCK A

GENERAL INPUT

A.3.6(MAXDCCL)	Set to zero if analysis only, KANALYZE(A.3.10) = 0. Otherwise, a typical value is 3 and LMSTR (A.3.3) will be one.
A.3.7 (MAXECCL)	(Same comments as for A.3.6) ... LMDSP (A.3.2) Will be one.
A.3.8 (IAREAS)	If set to one, then input values for THICK (E.1.10).
A.3.9 (INDMIN)	If set to zero, then input values for AEMNMM (A.4.1) for case of KANALYZE (A.3.10) = 1 and also input THKLAM (A.4.3) for case of NCOMP (C.1.4) $\neq$ 0. If set to one, then input THICKMN (E.1.11) and values for AEMNMM (A.4.1) and THKLAM (A.4.3) will be ignored.
A.3.10 (KANALYZE)	When an optimization is to be performed stress and/or deflection constraints will be present and LMTDSP (A.3.2) and/or LMTSTR (A.3.3) will be equal to one.
A.3.11 (MAXSZE)	Set to one, only if KANALYZE (A.3.10) is also one and then input THICKMX(3.1.12).
A.3.14 (MNLAYR)	If set to one, then input values for AEXMIN (F.6.1), AEYMIN (F.6.2), and AEXYMIN (F.6.3).
A.3.16 (IROUND)	If IROUND is set to one, then the standard sizes in the DATA statements in subroutine STDSIZ are used.
A.4.1 (AEMNMM)	If this is an optimization problem, KANALYZE (A.3.10) = 1, then this is the minimum allowable size of all the members.
A.4.2 (DINCR)	If displacement constraints are present, set DINCR to 1.1 as a representative value. This allows the displacement constraint algorithm a 10% margin in determining if the displacements exceed allowable values.
A.4.3 (THKLAM)	Usually set value to some multiple of ply thickness.
A.4.4 (SPRDF)	Usually set value between 0.5 and 0.8 so that shear thicknesses do not get too small during the earlier stress constraint cycles.
A.5.2 (NOI)	Recommend 10 iterations.

A.5.3 (TOLVEC)	Recommend 1.0E-5.
A.5.4 (NSTURM)	If value is set to one or greater, then the number of frequencies less than the FBDS (A.6.1) bounds can be determined. See Section 2.4.

INSTRUCTIONS

ADDRESS DATA BLOCK: B

TITLE: Coordinate Data

Inclusion Conditions: Required

Input Subroutine(s): INXYZ

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
B.1	B.1.1 (A15)	Coordi-nates (1-15)	Enter the literal "COORDINATES" or leave blank
	B.1.2 (I5)	NODES (16-20)	Total number of nodes in the finite element model.
	B.1.3 (I5)	MGEN (21-25)	Mesh generator flag =0 if the coordinates are to be input by the user =1 if the coordinates are to be read in the user subroutine MESG (Card Set B.1 contains one card)
B.2	B.2.1 (I5)	N (1-5)	Node Number
	B.2.2 (A1)	ISYS (6)	Reference coordinate system =(blank) for Cartesian coordinates =A for cylindrical coordinates R,O,Z =B for spherical coordinates R,O,o
	B.2.3 (I4)	NINCR* (7-10)	Increment for node generation
	B.2.4 (F10.3)	X1 (11-20)	First coordinate value for node N (B.2.1)
	B.2.5 (F10.3)	X2 (21-30)	Second coordinate value for node N (B.2.1)
	B.2.6 (F10.3)	X3* (31-40)	Third coordiante value for node N (B.2.1) (Card Set B.2 contains NODES(B.1.2) plus one cards. The last card is blank.)

*See Notes

NOTES ON DATA BLOCK B

COORDINATE DATA

B.2.3 (NINCR) A nonzero value of NINCR causes nodes to be equally spaced between the last and current nodes, with numbering equal to NINCR. As an example, for NINCR=4, the card set B-2

10		0.	0.	0.
26	4	8.	-24.	0.

is equivalent to

10	0.	0.	0.
14	2.	-6.	0.
18	4.	-12.	0.
22	6.	-18.	0.
26	8.	-24.	0.

B.2.6 (X3) If MM (A.3.1) = 2, then X3 = 0.0.

INSTRUCTIONS

ADDRESS DATA BLOCK: C

TITLE: General Element Data

Inclusion Conditions: Required

Input Subroutine(s): ELEMIN

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
C.1	C.1.1 (I5)	ITYPE* (1-5)	Always set to 3 - fixed
	C.1.2 (I5)	NMT (6-10)	Total number of material property sets, including composites
	C.1.3 (I5)	NELEM (11-15)	Total number of elements
	C.1.4 (I5)	NCOMP (16-20)	Total number of the NMT (C.1.2) material property sets which are composites. $0 \leq NCOMP \leq NMT(C.1.2)$
			(Card Set C.1 contains one card)

*See Notes

NOTES ON DATA BLOCK C

GENERAL ELEMENT DATA

C.1.1 (ITYPE)

ITYPE in ELEMIN is not the same as ITYPE in INP03. In subroutine ELEMIN the following variable assignments are made:

NMAT = NMT
MEMBS = NELEM
ISOTRN = NMAT - NCOMP = Number of isotropic material property sets.

INSTRUCTIONS

ADDRESS DATA BLOCK: D

TITLE: Materials Data

Inclusion Conditions: Required

Input Subroutine(s): INP03

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
D.1	D.1.1 (E10.0)	YOUNGM(I) (1-10)	If the material is isotropic, this is Young's modulus for the I-th material in psi. If the material is composite, this is Young's modulus for the I-th material in the zero degree direction in psi.
	D.1.2 (E10.0)	POISON(I) (11-20)	If the material is isotropic, this is Poisson's ratio for the I-th material. If the material is composite, this is Poisson's ratio for the transverse strain due to stress along the zero degree fiber direction for the I-th material.
	D.1.3 (E10.0)	RHO1(I) (21-30)	Weight density of the I-th material in lb/in^3.
	D.1.4 (E10.0)	ELASM(I) (31-40)	If the material is isotropic, then set this to zero. If the material is composite, then this is the elastic modulus transverse to the zero degree fiber direction of the I-th composite material in psi.
	D.1.5 (E10.0)	SHEARM(I) (41-50)	If the material is isotopic, then set this to zero. If the material is composite, then this is the shear modulus for the I-th composite material in psi. The isotropic material properties are ordered before the composite properties (Card Set D.1 contains NMT(C.1.2) cards.)

INSTRUCTIONS

ADDRESS DATA BLOCK: D

TITLE: Materials Data (continued)

Inclusion Conditions: Required

Input Subroutine(s): INP03

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
D.2	D.2.1 (E10.0)	ALSTRS(1) (1-10)	Tension stress allowable (strength) in the zero degree fiber direction in psi
	D.2.2 (E10.0)	ALSTRS(2) (11-20)	Compression stress allowable (strength) in the zero degree fiber direction in psi
	D.2.3 (E10.0)	ALSTRS(3) (21-30)	Tension stress allowable (strength) in the transverse fiber direction in psi
	D.2.4 (E10.0)	ALSTRS(4) (31-40)	Compression stress allowable (strength) in the transverse fiber direction in psi.
			The isotropic material properties are ordered before the composite properties
			(Card Set D.2 contains NMT(C.1.2) cards.)

INSTRUCTIONS

ADDRESS DATA BLOCK: E

TITLE: Connectivity Data

Inclusion Conditions: Required

Input Subroutine(s): INCONN

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
E.1	E.1.1 (I5)	IELNO (1-5)	Element number
	E.1.2 (I5)	ITYPE (6-10)	Element type: =1 for beam =2 for bar =3 for triangular membrane =4 for quadrilateral membrane =5 for shear panel
	E.1.3 (I5)	IPR* (11-15)	Material Property Set for this element
	E.1.4 (I5)	LAMFO* (16-20)	Fiber orientation parameter: =0 for isotropic element =1 for fiber orientations (0°,90°,±45°) with proportions (.25,.25,.50) =2 for fiber orientations (0°,90°) with proportion (.50, .50) =3 for fiber orientations ±45° with proportion 1.00 =4 for fiber orientations (0°,±45°) with proportions 1/3, 2/3) =5 for fiber orientations (90,±45°) with proportions (1/3, 2/3)

*See Notes

INSTRUCTIONS

ADDRESS DATA BLOCK: E

TITLE: Connectivity Data (continued)

Inclusion Conditions: Required

Input Subroutine(s): INCONN

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
	E.1.5 (I5)	KGEN* (21-25)	Node increment for element generation
	E.1.6 (I5)	NODE(1)* (26-30)	Local node number 1
	E.1.7 (I5)	NODE(2)* (31-35)	Local node number 2
	E.1.8 (I5)	NODE(3)* (36-40)	Local node number 3
	E.1.9 (I5)	NODE(4)* (41-45)	Local node number 4
	E.1.10 (E10.0)	THICK* (46-55)	Initial member size (thickness or area)
	E.1.11 (E10.0)	THICKMN (56-65)	Minimum member size (thickness or area)
	E.1.12 (E10.0)	THICKMX (66-75)	Maximum member size (thickness or area)
			(Card Set E.1 contains NELEM(C.1.3) plus two cards with the last two blanks when KGEN(E.1.5) is zero. See KGEN note also.)

*See Notes

NOTES ON DATA BLOCK E

CONNECTIVITY DATA

E.1.3 (IPR) The indexing scheme for material properties is established by the ordering of Data Block D.

E.1.4 (LAMFO) When specifying properties other than those shown, set IAREAS (A.3.8) equal to one.

E.1.5 (KGEN) A nonzero value of KGEN on the second card of a pair causes elements between the last and current elements to be automatically generated. With the exception of node numbers, all elements generated are assigned the same data as the current element. Local node numbers for the previous element are incremented by KGEN to generate each succeeding element. More than one element must be generated to use this feature; node numbers for the current element need not be given. As an example, the data

```
85    5    1    0    0    49    50    52    51    .0190
88
```

is equivalent to

```
85    5    1    0    0    49    50    52    51    .0190
86    5    1    0    0    51    52    54    53    .0190
87    5    1    0    0    53    54    56    55    .0190
88    5    1    0    0    55    56    58    57    .0190
```

E.1.6 (NODE(1)) Lowest node number for an element.

E.1.7 (NODE(2)) Next lowest node number for an element.

E.1.8 (NODE(3)) Leave blank for bar elements.

E.1.9 (NODE(4)) Leave blank for bar and triangular elements. For quadrilateral and shear panel elements, NODE(3) and NODE(4) are determined by continuing the direction defined by NODE(1) to NODE(2).

E.1.10 (THICK) For bars, THICK is the cross-sectional area. For all others, THICK is the element thickness.

INSTRUCTIONS

ADDRESS DATA BLOCK: F

TITLE: Fiber Orientation Data

Inclusion Conditions: LAYERD(A.3.12) ≠ 0

Input Subroutine(s): INLAYR

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
F.1	F.1.1 (F10.3)	XANG(I-1) (1-10)	The angle in degrees that the 0° fibers of the I-th element makes with the global x-axis
	F.1.2 (F10.3)	YANG(I-1) (11-20)	The angle in degrees that the 0° fibers of the I-th element makes with the global y-axis
	F.1.3 (F10.3)	ZANG(I-1) (21-30)	The angle in degrees that the 0° fibers of the I-th element makes with the global z-axis
		XANG(I) (31-40)	Next F.1.1
		YANG(I) (41-50)	Next F.1.2
		ZANG(I) (51-60)	Next F.1.3
		I=2,NELEM (C.1.3)	(Card Set F.1 contains six items per card until I equals NELEM(C.1.3)
			Include Card Set F.1 only if LAYERD(A.3.12)≠0 and INDANG(A.3.13)=0.

INSTRUCTIONS

ADDRESS DATA BLOCK: F

TITLE: Fiber Orientation Data (continued)

Inclusion Conditions:

Input Subroutine(s):

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
F.2	F.2.1 (F10.3)	XANG(I) (1-10, 11-20,..., (51-60) I=1,NELEM (C.1.3)	The angle in degrees that the 0° fibers of the I-th element makes with the local element coordinate system (Card Set F.2 contains six items per card until I equals NELEM(C.1.3)) Include Card Set F.2 only if LAYERD(A.3.12)≠0 and INDANG(A.3.13)=1.
F.3	F.3.1 (F10.3)	XA (1-10)	The angle the 0° fibers make with the global x-axis
	F.3.2 (F10.3)	YA (11-20)	The angle the 0° fibers make with the global y-axis
	F.3.3 (F10.3)	ZA (21-30)	The angle the 0° fibers make with the global z-axis (Card Set F.3 contains one card) Include Card Set F.3 only if LAYERD(A.3.12)≠0 and INDANG(A.3.13)=2

INSTRUCTIONS

ADDRESS DATA BLOCK: F

TITLE: Fiber Orientation Data (continued)

Inclusion Conditions:

Input Subroutine(s):

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
F.4	F.4.1 (F10.3)	AEX(I) (1-10, 11-20,..., 51-60) I=1,NELEM (C.1.3)	Proportion of fibers in the zero degree direction for the I-th element (Card Set F.4 contains six items per card until I equals NELEM(C.1.3)) Include Card Set F.4 only if LAYERD(A.3.12)≠0 and IAREAS≠0.
F.5	F.5.1 (F10.3)	AEY(I) (1-10, 11-20,..., 51-60) I=1,NELEM (C.1.3)	Proportion of fibers in the ninety degree direction for the I-th element (Card Set F.5 contains six items per card until I equals NELEM(C.1.3)) Include Card Set F.5 only if LAYERD(A.3.12)≠0 and IAREAS≠0.

INSTRUCTIONS

ADDRESS DATA BLOCK: F

TITLE: Fiber Orientation Data (continued)

Inclusion Conditions:

Input Subroutine(s):

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
F.6	F.6.1 (F10.3)	AEXMIN(I-1) (1-10)	Minimum total thickness of 0° layers
	F.6.2 (F10.3)	AEYMIN(I-1) (11-20)	Minimum total thickness of 90° layers
	F.6.3 (F10.3)	AEXYMIN(I-1) (21-30)	Minimum total thickness of ±45° layers
		AEXMIN(I) (31-40)	Next F.6.1
		AEYMIN(I) (41-50)	Next F.6.2
		AEXYMIN(I) (51-60)	Next F.6.3
		I=1,NELEM (C.1.3)	(Card Set F.6 contains six items per card: two sets of three properties; each element has one set of three) Include Card Set F.6 only if LAYERD(A.3.12)≠0 and MNLAYR(A.3.14) ≠0.

INSTRUCTIONS

ADDRESS DATA BLOCK: G

TITLE: Boundary Conditions

Inclusion Conditions: Required

Input Subroutine(s): GETBC

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
G.1	G.1.1 (I5)	NBC1* (1-5)	Number of Type 1 constraints
	G.1.2 (I5)	NBC2* (6-10)	Number of Type 2 constraints
			(Card Set G.1 contains one card.)
G.2	G.2.1 (I5)	N1* (1-5)	Beginning node number to be constrained
	G.2.2 (I5)	N2* (6-10)	Ending node number to be constrained
	G.2.3 (I5)	INCR (11-15)	Node number increment
			(Card Set G.2 contains one card)
			Include Card Set G.2 only if NBC1(G.1.1)≠0.
G.3	G.3.1 (I5)	JD(1)* (1-5)	Nodal components constrained
		JD(2)* (6-10)	(Card Set G.3 contains NBC1(G.1.1) cards)
		JD(3)* (11-15)	Include Card Set G.3 only if NBC1(G.1.1)≠0.

*See Notes

INSTRUCTIONS

ADDRESS DATA BLOCK: G

TITLE: Boundary Conditions (continued)

Inclusion Conditions:

Input Subroutine(s):

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
G.4	G.4.1 (I5)	JD(1)* (1-5) JD(2)* (6-10) JD(3)* (11-15)	Nodal components constrained (Card Set G.4 contains one card) Include Card Set G.4 only if NBC2(G.1.2)≠0.
G.5	G.5.1 (I5)	ND(I)* (1-5,6-10, ...,46-50)	Nodes constrained (Card Set G.5 contains at most ten items) Include Card Set G.5 only if NBC2(G.1.2)≠0. Card sets G.4 and G.5 are repeated in pairs NBC2(G.1.2) times.

*See Notes

NOTES ON BLOCK G

BOUNDARY CONDITIONS

G.1.1 (NBC1) A type 1 constraint is used to fix selected degrees of freedom over a specified range of nodes with a specified increment in node numbers.

G.1.2 (NBC2) A type 2 constraint is used to fix selected degrees of freedom over a list of nodes.

G.2.1 (N1) Type 1 constraint data to be entered NBC1 times.

G.3.1 (JD) Values of 1, 2, and 3 can be used. To fix u at a node (x component of deflection) include a 1 value. To fix v at a node (y component of deflection) include a 2 value. To fix w at a node (z component of deflection) include a 3 value. For example, make Card Set G.3

1 2 3

to fix all degrees of freedom. To fix only u use

1 0 0

G.4.1 (JD) Same comments as G.3.1.

G.5.1 (ND) Each of the nodes are constrained in the components specified by Card Set G.4.

INSTRUCTIONS

ADDRESS DATA BLOCK: H

TITLE: Loads Data

Inclusion Conditions: Required

Input Subroutine(s): INLOADS

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
H.1	H.1.1 (I5)	LOADS (1-5)	Number of loading conditions
	H.1.2 (I5)	NJLOADS (6-10, 11-15,etc)	Number of components in I-th loading condition for I=1,...,LOADS (Card Set H.1 contains one card with LOADS=1 values.)
H.2	H.2.1 (F10.3)	TEMP(I) (1-10)	Magnitude of I-th load
	H.2.2 (I5)	IM(I) (11-15)	Direction of I-th load =1 for x direction =2 for y direction =3 for z direction
	H.2.3 (I5)	JN(I) (16-20)	Node number for Ith load
		TEMP(I+1) (21-30)	Next TEMP
		IM(I+1) (31-35)	Next IM
		JM(I+1) (36-40)	Next JM
		TEMP(I+2) (41-50)	Next TEMP
		IM(I+2) (51-55)	Next IM
		JM(I+2) (56-60)	Next JM
			(Card Set H.2 is repeated so that a total of NJLOADS terms are entered. Process is repeated for each load case until all LOADS (H.1.1) cases are input.)

INSTRUCTIONS

ADDRESS DATA BLOCK: I

TITLE: Displacement Constraint Data

Inclusion Conditions: KANALYZE(A.3.10)=1, MAXDCCL(A.3.6)≠0.

Input Subroutine(s): INDSPL

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
I.1	I.1.1 (F10.3)	DEFMAX(1) (1-10)	Displacement constraint for x-direction
	I.1.2 (F10.3)	DEFMAX(2) (11-20)	Displacement constraint for y-direction
	I.1.3 (F10.3)	DEFMAX(3) (21-30)	Displacement constraint for z-direction (Leave blank if MM(A.3.1)=2.
			Include only if LMTDSP=1 (Card Set I.1 contains one card)
I.2	I.2.1 (I5)	KH (1-5)	Number of displacement constraints
			Include only if LMTDSP=2. (Card I.2 contains only one card)

INSTRUCTIONS

ADDRESS DATA BLOCK: I (continued)

TITLE: Displacement Constraint Data

Inclusion Conditions: KANALYZE(A.3.10)=1, MAXDCCL(A.3.6)≠0.

Input Subroutine(s): INDSPL

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
I.3	I.3.1 (F10.3)	TEMP(I) (1-10)	Magnitude of I-th displacement constraint
	I.3.2 (I5)	IM(I) (11-15)	Direction of I-th displacement constraint =1 for x direction =2 for y direction =3 for z direction
	I.3.3	JM(J) (16-20)	Node number of I-th displacement constraint
		TEMP(I+1) (21-30)	Next TEMP
		IM(I+1) (31-35)	Next IM
		JM(I+1) (36-40)	
		TEMP(I+2) (41-50)	Next TEMP
		IM(I+2) (51-55)	Next IM
		JM(I+2) (56-60)	Next JM
			(Card Set I.3 is repeated so that a total of KH(I.2.1) values for TEMP, IM, and JM are entered.)

INSTRUCTIONS

ADDRESS DATA BLOCK: J

TITLE: Damage Data

Inclusion Conditions: Include if NDMGCAS(A.3.4)≠0

Input Subroutine(s): INDAMAGE

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
J.1	J.1.1	TOLDMGD	Damage iteration tolerance
	(F10.3)	(1-10)	(0.1 is a typical value)
			Approximate reanalysis solutions will be accurate to 0.1.
	J.1.2 (I5)	MAXDDIT (11-15)	Maximum number of reanalysis iterations (10 is a typical value)
	J.1.3 (F10.3)	TOLDMGF (16-25)	Mode shape iteration tolerance (0.1 is a typical value)
	J.1.4 (I5)	MAXDFIT (26-30)	Maximum number of mode shape iterations (10 is a typical value)
			Items J.1.1 and J.1.2 determine the accuracy of intermediate approximate analyses for static loads. Items J.1.3 and J.1.4 apply only if mode shapes are to be computed.
			(Card Set J.1 contains 1 card.)

INSTRUCTIONS

ADDRESS DATA BLOCK: J (continued)

TITLE: Damage Data

Inclusion Conditions: Include if NDMGCAS(A.3.4)≠0

Input Subroutine(s): INDAMAGE

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
J.2	J.2.1 (I5)	NDMG(J) (1-5)	Number of elements in damage case, J. (Card Set J.2 contains one card. Card Sets J.2 and J.3 are repeated in pairs for J=1,...,NDMGCAS (A.3.4).) be accurate to 0.1.
J.3	J.3.1 (I5)	IDNG(J,I) (1-5)	I-th elements in J-th damage case
	J.3.2 (F10.3)	DKF (6-15)	Stiffness damage factor between 0 and 1 where 0. = no damage 1. = total damage (elements removed)
	J.3.3 (F10.3)	DMF (16-25)	Mass damage factor (same comments as for J.3.2) (Card Set J.3 contains NDMG(J) cards.)

INSTRUCTIONS

ADDRESS DATA BLOCK: K

TITLE: Lumped Mass Data

Inclusion Conditions: NLMPMSS(A.3.5)≠0

Input Subroutine(s): INPMSS

CARD SET	ITEM (FORMAT)	VARIABLE (COL)	DESCRIPTION (NUMBER OF CARDS PER SET)
K.1	K.1.1	LMNODE(I)	Node location of I-th lumped mass
	(I5)	(1-5)	
	K.1.2 (F10.3)	WLMPMSS(I) (6-15)	Value of I-th lumped mass, in lb.
			(Card Set K.1 contains NLMPMSS(A.3.5) cards.)

SECTION 5
PROGRAM OUTPUT

Output of the ADDRESS program is very similar in nature to the OPTSTAT [3] program. Formats have been changed to some extent and additional information on damage cases is printed. Output for a typical static optimization run is as follows:

1. Input Data Echo. Fifty lines of data are printed per page with header and trailer numbers to help check the column locations of the data. The title card information is printed on all pages of the output, together with a page number.

2. General Data. Card Set A.3 information is presented with explanations of the numerical values.

3. Nodal Coordinates. Data is always printed in Cartesian coordinates even though another system may be used for input.

4. Materials Data. Properties from Card Set D.1 and the allowables from Card Set D.2 are printed for each material.

5. Element Connectivity. Element number (ELEM), type (ITYPE), material code (MATL), fiber orientation parameter (LAM), local node number, and member sizes are printed, based on the Card Set E.1 data.

6. Composite Element Data. If the model contains composite elements the fiber orientation data from Card Sets F.1, F.4, F.5, and F.6 are printed.

7. Boundary Conditions. A summary of the type 1 and type 2 boundary conditions (see notes for Card Sets G.1 through G.5 of Section 4) is printed. For the current library of elements, components 4 through 10 will always be printed as zeroes.

8. Load Summary. For each load case the net loads, (FX, FY, FZ) in the coordinate directions are printed, together with their respective elements (MX, MY, MZ).

9. Damage Data. Data from Card Sets J.1, J.2, and J.3 are printed for each damage case.

10. Population Information.[3] Output from Subroutine POP concerning the distribution of elements in the stiffness matrix. This information is generated before the stiffness matrix of the structure is assembled.

(a) Gross Population = total number of elements in the upper triangle of the matrix.

Apparent Population = actual number of elements considered as nonzero by a given solution scheme. Thus the apparent population represents the number of storage locations required for the stiffness matrix.

(b) Starting Row Numbers for each column - the number of the row where the first nonzero element occurs in each column.

(c) Numbers of Diagonal Elements in Single Array Stiffness Matrix. For each Column I the actual number of elements, ID(I), in the upper triangular matrix up to and including that column, i.e.,

$$ID(I) = \frac{I(I+3)}{2} - \sum_{j=1}^{n} b_j$$

where b_j is the row number given for Column I in (b). Thus for the last column, ILAST,

ID(ILAST) = Apparent Population

11. Relative Design Data. After the static analysis is performed, the relative sizes of the elements are printed. The maximum relative size is 1.0 and all other sizes are given relative to this largest size. Next, a summary of the time for the analysis is printed. This is broken down into time for the

matrix assembly, decomposition, and forward and backward substitution.

12. Damage Reanalysis Error Summary. If damage cases are included, then the analysis iteration error summary is printed for each case. The norm of the vector error is printed with the load case index in parentheses. Error information on the extrapolation process is also printed. Total time required to analyze each damage case is indicated. This should be less than the time for the analysis of Item 11 above for the reanalysis to be cost-effective.

13. Initial Scale Factor (BASEA). This is defined in subroutine SCALE to be .01 times the square root of the quotient of STRMAX and ENGCAP. The variable STRMAX is the maximum over all load vectors of the inner product of the load and displacement vectors. In STIFFK, ENGCAP is defined as the sum over all elements of AE times ELENGTH where AE is the member size and ELENGTH is length (for bars) or area (for membranes and shear panels). The initial BASEA is used in PREPAR to scale the stress allowables. These scaled allowables, ALS, are used in STRCON to compute the element stress ratios, ESRTIO.

14. Scaling Factors. If a stress ratio for an element in some load/damage case combination exceeds 1.0, its maximum value is printed. The listing of corresponding element numbers that is printed beside these exceedance values tells the user which elements are overstressed for the design whose relative member sizes are scaled by the initial scaling factor of Item 13. The product of the initial scaling factor and the critical element scaling factors gives the new BASEA which is printed next with the header, "Scaling factor to satisfy stress constr."

15. Summary of Current Cycle. Current cycle number of this resizing is printed, together with a weights summary.

16. Resizing Continues. Items 11 through 15 are repeated until one of the following happens:

(a) All energy and displacement cycles have been completed as specified in Card Set A.3;

(b) The weight goes up in an energy cycle;

(c) The weight doubles in a displacement cycle.

17. Final Design Stress Summary. Stresses, member sizes, element numbers, and ESRATIO's are printed. The column of ESRATIO'S are the last set of stress ratios computed (see Item 13). They are ordered so that the undamaged load cases come first, followed by the damage case information by load case. Example: 2 load cases, 2 damages,

	ELMT	ESRATIO	
element no.	5	.91162E-01 LC1,	no damage
		.74404E-01 LC2,	no damage
		.12544E+00 LC1,	DC1
		.10595E+00 LC2,	DC1
		.91743E-01 LC1,	DC2
		.75008E-01 LC2,	DC2

LC = load case, DC = damage case.

18. Final Design Deflection Summary. Deflections are printed by node in the three coordinate directions. Applied loads are also printed in the same format as the ESRATIO's above in Item 17.

Additional output is written to local files 8 and 99. The subroutines beginning with "PP" write information to one or both of these files for post-processing.

SECTION 6
PLOTTING UTILITIES

Two interactive computer graphics processors for ADDRESS have been developed. The first, ADDPLOT, may be used for either preanalysis model verification or for postanalysis geometry displays. The second, CONTOR, is a contour plotter which displays patterns of constant displacement, thickness, strain energy density and stress-ratio data. Both plotters are command driven and contain many options to facilitate manipulation of results and geometry. Although similar in their usage, these programs shall be discussed separately.

The plotting utilities ADDPLOT and CONTOR offer great versatility in displaying results and in verifying model geometry. Experience has shown that the best way to become familiar with the various commands within these programs is to simply try a command to see its effect. Each command entered has the effect of modifying the appearance of the next plot, and options may be switched on and off quite easily. A short session with ADDPLOT or CONTOR, together with generous use of the HELP and SUMMary commands, is usually the best introduction to the use of these programs. Note that, in both programs, sufficient default values are defined that the single command DRAW will produce a first plot for use as a starting point.

6.1 GEOMETRY PLOTTING

ADDPLOT will accept input from either an ADDRESS input file or from a postanalysis (DPOST) output file. After starting the program, the user may enter commands, most of which have the effect of altering the appearance of a subsequent plot. A plot is drawn only when the command DRAW is entered; therefore, one can change plotting parameters as often as desired, with only the latest settings being retained. Reasonable default values are set for all parameters upon entering ADDPLOT, so that the command to DRAW always produces a plot.

Commands recognized by ADDPLOT are listed and described briefly below. Unless noted, all commands apply to both pre- and postanalysis plotting.

AXES - Selects/deselects drawing and labelling of the coordinate axes. (The default is no axes drawn.)

BARS - Selects/deselects whether or not the truss elements are plotted. (The default is to plot all trusses.)

CHARACTER - Selects the ADE character size to be used when labelling nodes and elements. The sizes range from 1 to 4 with 4 being the smallest. (The default is character size 4.)

CLIP - Defines the "clip factor", defined as C/P, where C is the distance from the viewer to the clipping plane, and P is the distance from the viewer to the site position. Points between the viewer and the clipping plane are not plotted. This command is rarely used. (The default is C/P = 0.01.)

CUBE - Allows user to select the region in space which will be plotted on the screen. The program automatically sets the cube to the model limits. This command is sometimes used to view only a portion of the model on the screen. For most applications, "ZOOM" is easier to use. (The default cube is the model limits.)

DEFAULT - Resets all plotting parameters to their default values. The default values of all the commands are given along with their description.

DEFORM - Selects the solution increment to be plotted for postanalysis geometry plots. The deformed geometry may be plotted alone or superimposed on the original

model geometry. Displacements may be scaled. (The default is to draw undeformed geometry only.)

DRAW - Displays a plot on the screen according to the present plotting parameters.

ELEMENTS - User may select which elements of the model are plotted with this command. The user can select either random elements or ranges of elements. (The default is to plot all elements of the model.)

EYE - Defines the user eye position in space. Allows the user to view a model from any point or orientation. (The default eye position is X=100, Y=100, Z=100.)

HELP - Lists all of the available commands along with a brief explanation of them.

LABELS - Allows node and/or element numbers to be displayed. Also provides for highlighting nodes by drawing a small circle around them. (The default is no labelling or highlighting.)

PROJECTION - Toggles between perspective and orthogonal projection of the display. (The default is perspective.)

REFLECT - Provides for reflecting a plot across a coordinate plane. Both the original model and the reflection are plotted. (The default is no reflection.)

ROTATE - Provides for rotation around a coordinate axis. Rotations are specified in degrees and the direction is given by the right hand rule. (The default is no rotation.)

SCALE - ADDPLOT scales all plots automatically to be properly proportioned when appearing on the screen. With this command the user can toggle this scaling on and off. (The default is proper scaling.)

SHRINK - Reduces each element by a shrink factor to produce an "exploded" view of the model. The shrink factor must be between zero (no shrinkage) and one (all elements shrink to points). (The default is no shrinkage.)

SITE - Allows the user to select which geometric point will appear approximately at the center of the screen. (The default site position is the centroid of the model.)

STOP - Terminates execution of ADDPLOT.

SUMMARY - Lists all of the present plotting parameters.

TIME - Gives the cpu time used since logging onto the system.

TRANSLATE - Provides for linear translation of the model in space. Frequently used to move the model near the origin so that the "AXES" may be plotted. (The default is no translation.)

SURFACE - Selects all of the panel (membrane, shear panel) elements connected to a single surface of the model. Normally this option is used to obtain a simplified geometry plot containing, for example, a single surface of a wing structure. This command should not be used simultaneously with "ELEM" as results are unpredictable. (The default is no surface plotting.)

VERTICAL - Selects which coordinate axis shall appear vertically on the screen. (The default vertical axis is Z.)

ZOOM - Allows for "zooming" in upon a portion of the model. The area defined by the user is expanded to fill the screen on the next plot. (The default is no zoom.)

Figures 7 and 8 show two sample plots produced using ADDPLOT. The first example gives a simple initial geometry plot of the upper surface of a wing along with a "SUMMARY" of the parameters used to generate the plot. The second example gives a postanalysis deformed geometry plot of the same model. Again a summary of the plotting options is given.

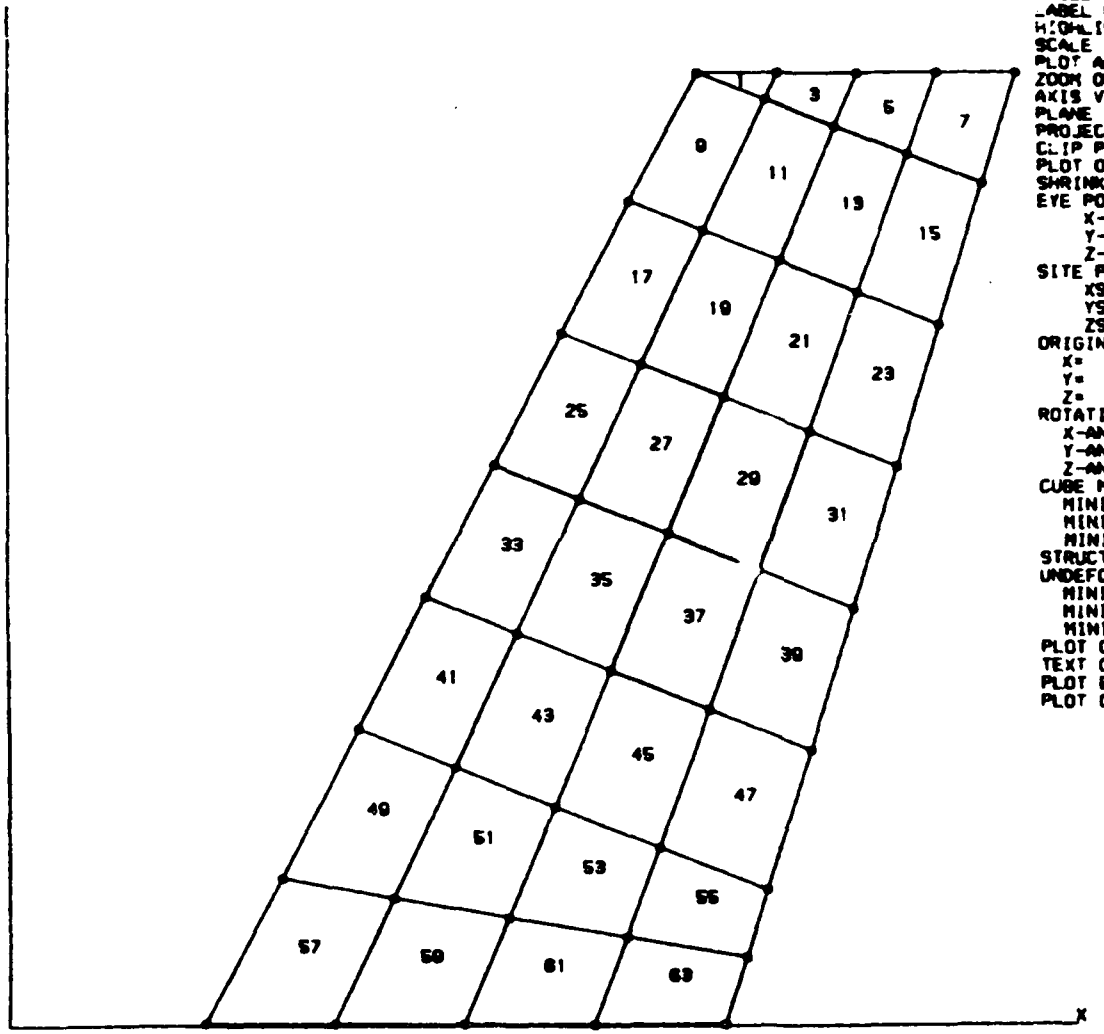
6.2 CONTOUR PLOTTING

CONTOR produces plots showing model geometry and contour lines for any of several geometry and response variables, including:

* X displacements
* Y displacements
* Z displacements
* Displacement magnitudes
* Elemental thicknesses
* Stress ratios
* Strain energy densities

CONTOR uses many of the same commands as ADDPLOT for controlling the appearance of a plot; several additional commands are provided

ADDPLOT Output:



Selected Plot Parameters:

```
LABEL AXES. . . . . : YES
LABEL ELEMENTS. . . . : YES
LABEL NODES . . . . . : NO
HIGHLIGHT NODES . . . : YES
SCALE PLOT. . . . . : YES
PLOT ALL ELEMENTS . . : NO
ZOOM OPTION . . . . . : NO
AXIS VERTICAL . . . . .Y-AXIS
PLANE REFLECTED . . . .NONE
PROJECTION TYPE . . . .PERSPECTIVE
CLIP PLANE FACTOR . . .      0.0100
PLOT OPTION . . . . . .UNDEFORMED
SHRINK FACTOR . . . . .      0.0000
EYE POSITION:
     X-EYE=          0.0000
     Y-EYE=          0.0000
     Z-EYE=      10000.0000
SITE POSITION:
     XSITE=         50.1505
     YSITE=         40.5000
     ZSITE=          0.3000
ORIGIN TRANSLATION:
   X=          0.0000
   Y=          0.0000
   Z=          0.0000
ROTATION:
   X-ANGLE=          0.0000
   Y-ANGLE=          0.0000
   Z-ANGLE=          0.0000
CUBE MINIMA AND MAXIMA:
   MINIMUM X     1.000000E+01 MAXIMUM X     9.283000E+01
   MINIMUM Y     0.000000E+00 MAXIMUM Y     0.000000E+01
   MINIMUM Z    -3.000000E+00 MAXIMUM Z     3.000000E+00
STRUCTURE NUMBER. . . .  1
UNDEFORMED MINIMA AND MAXIMA:
   MINIMUM X     1.000000E+01 MAXIMUM X     9.283000E+01
   MINIMUM Y     0.000000E+00 MAXIMUM Y     0.000000E+01
   MINIMUM Z    -3.000000E+00 MAXIMUM Z     3.000000E+00
PLOT CHAR. SIZE . . . .  4
TEXT CHAR. SIZE . . . .  4
PLOT BARS . . . . . : NO
PLOT ONLY 1 SURFACE . . NO
```

Figure 7. Undeformed Wing Upper Surface Plot.

ADDPLOT Output:

Selected Plot Parameters:

```
LABEL AXES. . . . . : NO
LABEL ELEMENTS. . . . . : NO
LABEL NODES . . . . . : NO
HIGHLIGHT NODES . . . . : NO
SCALE PLOT. . . . . : YES
PLOT ALL ELEMENTS . . . : NO
ZOOM OPTION . . . . . : YES
AXIS VERTICAL . . . . . :Z-AXIS
PLANE REFLECTED . . . . :NONE
PROJECTION TYPE . . . . :PERSPECTIVE
CLIP PLANE FACTOR . . . :      0.0100
PLOT OPTION . . . . . :UNDEF&DEF
SHRINK FACTOR . . . . .      0.0000
EYE POSITION:
    X-EYE=      -1000.0000
    Y-EYE=          0.0000
    Z-EYE=        100.0000
SITE POSITION:
    XSITE=         50.2887
    YSITE=         40.0225
    ZSITE=         11.1083
ORIGIN TRANSLATION:
  X=          0.0000
  Y=          0.0000
  Z=          0.0000
ROTATION:
  X-ANGLE=          0.0000
  Y-ANGLE=          0.0000
  Z-ANGLE=          0.0000
CUBE MINIMA AND MAXIMA:
  MINIMUM X     1.000000E+01 MAXIMUM X    9.98304E+01
  MINIMUM Y     0.000000E+00 MAXIMUM Y    8.00853E+01
  MINIMUM Z    -5.000000E+00 MAXIMUM Z    2.201508E+01
STRUCTURE NUMBER. . . .     1
UNDEFORMED MINIMA AND MAXIMA:
  MINIMUM X     1.000000E+01 MAXIMUM X    9.98304E+01
  MINIMUM Y     0.000000E+00 MAXIMUM Y    8.00853E+01
  MINIMUM Z    -5.000000E+00 MAXIMUM Z    2.201508E+01
DEFORMED MINIMA AND MAXIMA:
  MINIMUM X     1.000000E+01 MAXIMUM X    9.98304E+01
  MINIMUM Y     0.000000E+00 MAXIMUM Y    8.00853E+01
  MINIMUM Z    -5.000000E+00 MAXIMUM Z    2.201508E+01
INCREMENT NUMBER. . . .     1
DEFORM SCALE. . . . . :     1.0000
PLOT CHAR. SIZE . . . . :     4
TEXT CHAR. SIZE . . . . :     4
PLOT BARS . . . . . : YES
PLOT ONLY 1 SURFACE . . : NO
```

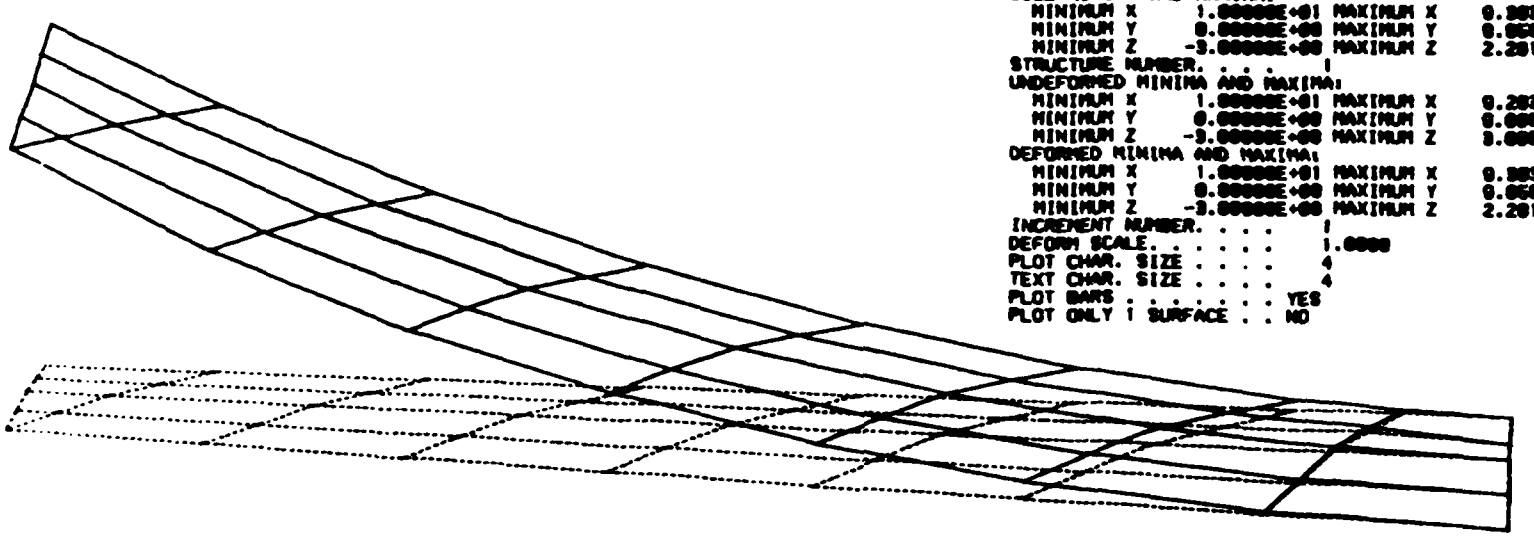


Figure 8. Deformed Wing Upper Surface Plot.

to define the data to be used for contour plotting. All of the CONTOR commands are listed and described below for easy reference.

CLIP - Defines the "clip factor", defined as C/P, where C is the distance from the viewer to the clipping plane, and P is the distance from the viewer to the site position. Points between the viewer and the clipping plane are not plotted. This command is rarely used. (The default is C/P = 0.01.)

CORE - Selects contour line values, the contour label placement, the contour labelling element frequency, and the contour labelling character size. The contouring values specify the range for which contours are drawn. Contour labels may be drawn in any of four character sizes, ranging from 1 to 4, with 4 being the smallest. Also, the user may specify where within an element a label will appear and how often contours are labelled. (The default is to draw 10 contour levels from the minimum to the maximum, with no labelling done.)

CONL - Does everything that "CORE" does except select the contouring values. May be used to label contours after the desired number of contour levels has been chosen.

CUBE - Allows user to select the region in space which will be plotted on the screen. The program automatically sets the cube to the model limits. This command is sometimes used to view only a portion of the model on the screen. For most applications, "ZOOM" is easier to use. (The default cube is the model limits.)

DEFO - Selects whether the plot shall contain the original geometry, the deformed geometry, or both. Also allows for scaling of displacements. Usually used in conjunction with "NINC". (The default is the original geometry.)

DRAW - Displays a plot on the screen according to the present plotting parameters.

ELEM - User may select which elements of the model are plotted with this command. The user can select either random elements or ranges of elements. (The default is to plot all elements of the model.)

EXIT - Terminates execution of CONTOR.

EXPL - Reduces each element by a shrink factor to produce an "exploded" view of the model. The shrink factor must be between zero (no shrinkage) and one (all elements shrink to points). (The default is no shrinkage.)

EYEP - Defines the user eye position in space. Allows the user to view a model from any point or orientation. (The default eye position is X=100, Y=100, Z=100.)

HELP - Lists all of the available commands along with a brief explanation of them.

LABE - Selects/deselects whether or not the element numbers are displayed. (The default is no element labelling.)

LAXS - Selects/deselects drawing and labelling of the coordinate axes. (The default is no axes drawn.)

MDRA - Allows the user to draw several solutions in succession using the same plotting parameters.

MOVI - Retains the plotting scale for all increments plotted so that differences from plot to plot may be detected.

NDAT - Lets the user select among the many possible solution variables for contour plotting. (The default contouring option is displacement magnitudes.)

NINC - Retrieves a new solution increment, load case, or mode shape from the data file. (The default is to retrieve the first solution on the file.)

NODE - Selects/deselects labelling of node numbers. (The default is no node labelling.)

PROJ - Toggles between perspective and orthogonal projections in the display. (The default is perspective.)

REFL - Provides for reflecting a plot across a coordinate plane. Both the original model and the reflection are plotted. (The default is no reflection.)

RESE - Sets all plotting parameters to their default values. The default values are given in parentheses in the command descriptions.

SITE - Allows the user to select which geometric point will appear approximately at the center of the screen. (The default site position is the centroid of the model.)

STEP - Selects the step size to be used in the contour line integration. A smaller step size makes for more accurate contours but requires more processing time. (The default step size is 0.05.)

SUBT - Allows the user to place a subtitle in the title block of a plot if "TITL" has been selected. (The default is no subtitle.)

SUMM - Lists all of the present plotting parameters.

TIME - Gives the cpu time used since logging onto the system.

TITL - Frames the plot and gives pertinent information about what the plot represents. (The default is no title block.)

TYPE - The user may select only the membranes, shear panels, bars, or all elements for plotting. (The default is to plot the membrane elements only.)

VERT - Selects which coordinate axis shall appear vertically on the screen. (The default vertical axis is Z.)

ZOOM - Allows for "zooming" in upon a portion of the model. The area defined by the user is expanded to fill the screen on the next plot. (The default is no zoom.)

Figure 9 shows an example of plotting with CONTOR. The contour lines show equal displacement magnitude levels. These contours are drawn on the deformed upper surface of a wing model. The original position of the wing is also drawn with a solid line. A SUMMary of the options used to generate this picture is also given.

CONTOUR Output

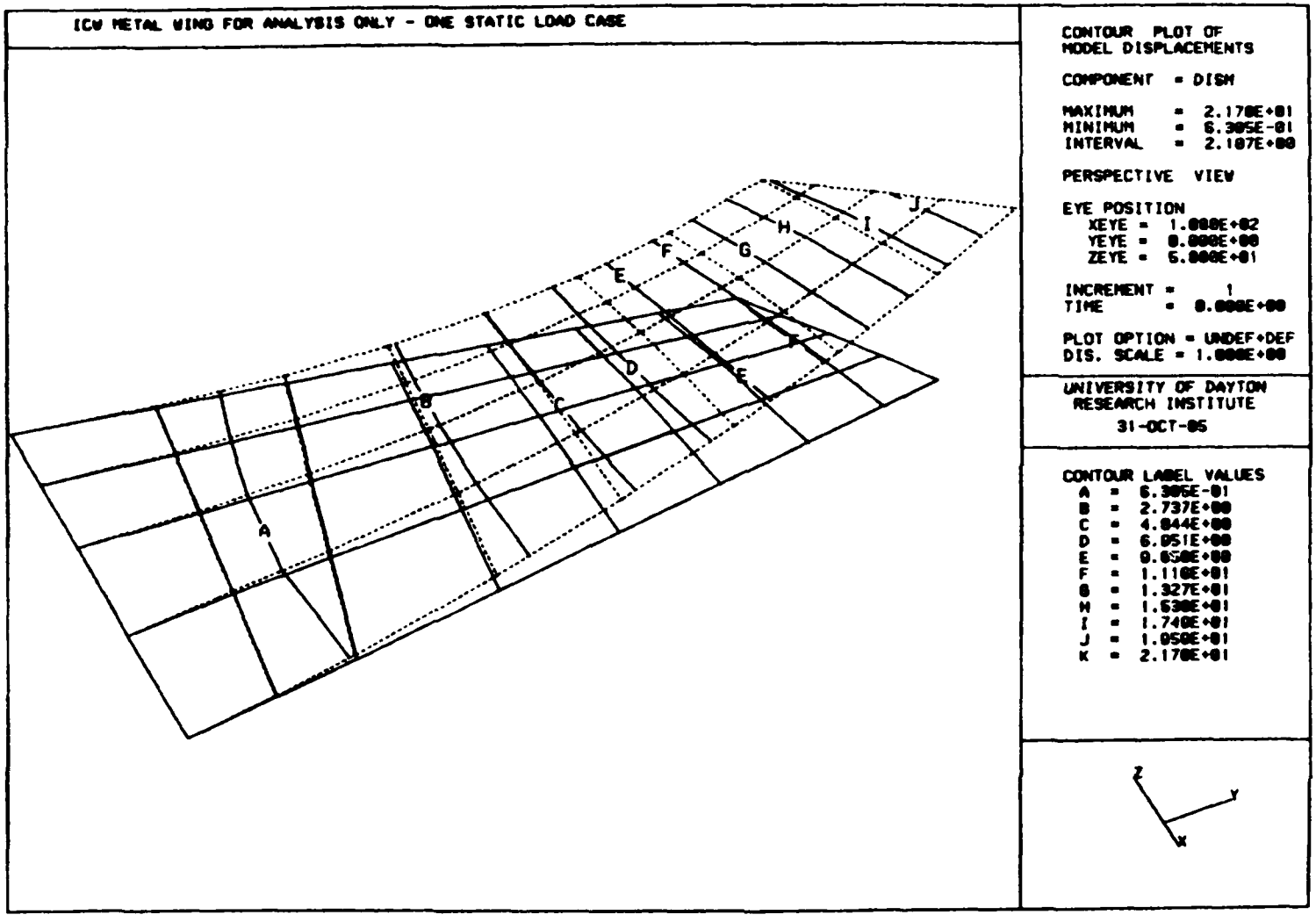


Figure 9. Deformation Contour Plot for Upper Wing Surface.

SECTION 7
COMMAND PROCEDURES FOR VAX-11/780 OPERATION

A number of command procedures have been developed to make installation, setup, and execution of ADDRESS particularly simple on the VAX-11/780 computer. This Section describes the most important procedures used to set up and run ADDRESS on the VAX.

7.1 ADDRESS INSTALLATION

The procedure for installing ADDRESS is largely automated, and is performed primarily under control of the command procedure INSTALL.COM. Installation of the program involves the following three steps:

(1) Create a separate directory for ADDRESS; for example:

```
$ CREATE/DIR [username.ADDRESS]
$ SET DEFAULT [username.ADDRESS]
```

This directory will contain all ADDRESS files, with the exception of user data. The string "username" must be replaced by a valid directory name.

(2) Mount the ADDRESS delivery tape and copy all files to the ADDRESS directory; for instance:

```
$ MOUNT/DENSITY=1600 device: ADDRES AA
$ COPY device:*.* *.*;*
```

Here "device:" is the device name for the tape drive on which the tape is to be mounted.

(3) Invoke the installation procedure INSTALL.COM:

```
$ @INSTALL
```

This step must be performed interactively. When the necessary files are prepared, the INSTALL procedure automatically begins the ADDRESS setup and compilation session described in subsection 7.2.

When the installation procedure is completed, ADDRESS will have been compiled and linked, and is ready to use.

Note that, when ADDRESS or any of its companion programs are used, it is necessary to provide the complete file specification for these programs, e.g., [username.ADDRESS]ADDRESS. To avoid this inconvenience, an additional command procedure, DEFINE.COM, is provided. Users of ADDRESS may wish to execute this command file as part of their normal login procedure. That is, ADDRESS users may add the statement

```
$ @[username.ADDRESS]DEFINE
```

to the LOGIN.COM file on their user i.d. As before, "username" must be replaced by the actual main directory under which ADDRESS has been installed. With this addition to a user's login procedure, it will no longer be necessary to provide complete directory information when executing ADDRESS and its companion programs.

7.2 ADDRESS SETUP AND COMPILATION

As described in Subsection 2.1, all of the major data arrays in ADDRESS are now adjustable. Problem size limits are largely open-ended, the only real limitation being the allowable task size and the disk capacity of the computer system. Maximum problem size parameters are established by executing a simple interactive setup procedure, which creates and compiles a version of ADDRESS with the specified problem size capacity.

A sample ADDRESS setup session is listed below. Note that the files ADDCOM.DAT, ADDIFG.EXE, ADDRESS.FOR, and SETUP.COM, all of which are provided on the ADDRESS delivery tape, should be present before executing SETUP. The setup procedure takes only a minute or two to complete, and is executed interactively. Data and commands entered at the terminal keyboard are underlined in the session listing below. Note that the listings below have been compressed to fit within page width restictions.

```
$ DIR/SIZE/DATE/PROT

Directory DRA2:[ADDRESS.EXAMPLE]

ADDCOM.DAT;3              18  22-JUL-1985 02:58  (R,RWE,,)
ADDIFG.EXE;3              12  13-JUL-1985 02:05  (R,RWE,,)
ADDIFG.FOR;4              21  13-JUL-1985 02:05  (R,RWE,,)
ADDRESS.FOR;32           429  30-SEP-1985 03:09  (R,RWE,,)
SETUP.COM;3                7  13-JUL-1985 02:10  (R,RWE,,)

Total of 5 files, 487 blocks.

$ @SETUP

**********************************************************************
**                                                                  **
** Beginning ADDRESS storage allocation/compilation/link edit       **
**                                                                  **
** This procedure builds a new executable copy of ADDRESS, in       **
** three major steps:                                               **
**                                                                  **
**       1.  Definition of program storage capacity                 **
**       2.  FORTRAN compilation                                    **
**       3.  Linking                                                **
**                                                                  **
** You will be requested to enter values for several variables      **
** which control the maximum size problem that ADDRESS will be      **
** able to solve.  The remainder of the procedure is automatic      **
** and will take only a few minutes to complete.                    **
**                                                                  **
**********************************************************************

  Press any key to continue, or type STOP to exit...: <CR>


      ************************************************
      *                                              *
      *    ADDIFG: ADDRESS Include File Generator    *
      *                                              *
      ************************************************

   Data will be requested below to set the problem size
   capacity of ADDRESS. For each item of data, minimum,
   maximum and default values will be displayed.  Enter
   the appropriate value (or an empty carriage return,
   which causes the default value to be used).
```

```
Parameter Description          Minimum Maximum Default VALUE
------------------------------ ------- ------- ------- ------
Max. constrained D.O.F.             50    none     100    200
Max. matrix partitions              50    none     100    100
Max. damage conditions               2    none       2    5
Max. elements per damage case       10    none      25    50
Max. degrees of freedom            300    none    1500    5000
Max. finite elements               100    none    1000    3000
Max. load cases (or modes)           6    none       6    10
Max. lumped masses                  10    none      10    10
Max. node points                   100    none     500    3000
Max. direct file records           500    none     500    500
Work space for matrix storage    10000    none   50000    200000

Writing ADDRESS common files...
File generation complete.
Job ADDRESS_COMPILATION (queue SYS$BATCH, entry 932) started
on SYS$BATCH

Batch procedure has been submitted to compile and link ADDRESS;
Executable program will be created on this directory as
ADDRESS.EXE.

<<<<<<<<<<<<<<<<<< ADDRESS Setup Complete >>>>>>>>>>>>>>>>>>

$
Job ADDRESS_COMPILATION (queue SYS$BATCH, entry 932) completed
Job ADDRESS_COMPILATION (queue LPA0, entry 933) completed
```

(end of listing)

When the system message(s) announcing completion of the ADDRESS compilation appear, two new files have been created: the first of these is ADDRESS.EXE, the executable version of ADDRESS. The second file is a log file for the SETUP session, which summarizes the limits set in the most recent version of ADDRESS. This file is always called ADDIFG.LOG. A listing of a typical log file is presented below:

```
ADDRESS Setup Performed on 30-SEP-85 at 03:10:54

Parameter Description          Minimum Maximum Default VALUE
------------------------------ ------- ------- ------- ------
Max. constrained D.O.F.             50    none     100    100
Max. matrix partitions              50    none     100    100
Max. damage conditions               2    none       2      2
Max. elements per damage case       10    none      25     25
Max. degrees of freedom            300    none    1500   1500
Max. finite elements               100    none    1000   1000
Max. load cases (or modes)           6    none       6      6
Max. lumped masses                  10    none      10     10
Max. node points                   100    none     500    500
Max. direct file records           500    none     500    500
Work space for matrix storage    10000    none   50000  50000
```

(end of listing)

The setup log file provides a reminder of the parameter settings for ADDRESS. If a number of users have access to the procedure for resetting ADDRESS storage parameters, it may be advisable to review the current values prior to performing a large analysis.

7.3 ADDRESS EXECUTION

Problem data for ADDRESS must be entered in a data file according to the instructions in Section 4 of this report. A utility procedure, ADDRUN, is provided with ADDRESS to assist in preparing the command file for a batch analysis run. If the DEFINE procedure has been executed as part of a user's login procedure, the VMS command "@ADDRUN" will invoke this procedure. A typical session with the ADDRUN utility is listed in full below.

```
$ @ADDRUN

  <<<<<<<<<<<<<< Begin ADDRESS Batch Run SETUP >>>>>>>>>>>>>>

  The following data files are in the current directory:

Directory DRA2:[ADDRESS.RAB.INS]

ADDEX1.DAT;3              18  22-JUL-1985 02:58  (R,RWED,RWED,)
EXAMPLE.DAT;1             32  27-AUG-1985 11:49  (R,RWED,RWED,)

Total of 2 files, 50 blocks.

Enter data file name (without '.DAT'): EXAMPLE
```

```
  ADDRESS Batch Run Procedure:
------------------------------------------------------------

$ set def [ADDRESS.RAB.INS]
$ copy EXAMPLE.DAT ADDIN.DAT
$ run ADDRESS
$ del ADDIN.DAT;0
$ on error then continue
$ rename ADDOT.OUT EXAMPLE.OUT
$ rename DPOST.DAT EXAMPLE.DPO
$ rename MODESHAPE.DAT EXAMPLE.MDS
$ del ADDRESS.RUN;0
$ exit

Run log file will be written to [ADDRESS.RAB.INS]EXAMPLE.LOG

------------------------------------------------------------

Submit ADDRESS job as listed above (Y/N)? : Y

Job ADDRESS (queue SYS$BATCH, entry 1644) started on SYS$BATCH

  <<<<<<<<<<<<<<< ADDRESS Batch Run Entered >>>>>>>>>>>>>>>

$
(end of listing)
```

The ADDRUN procedure writes a command file, ADDRESS.RUN, which contains appropriate commands for executing ADDRESS with the specified data (EXAMPLE.DAT in the sample above). If the user answers "Yes" to the final prompt, a batch run is submitted automatically. If the user answers "No", the procedure exits, and the command file ADDRESS.RUN remains on the current default directory. This file can be edited if necessary, and then submitted for execution using the VMS command SUBMIT; for instance, if the job is to run immediately after midnight, one might enter:

```
$ Submit/Log=ADDRESS.LOG/After=Tomorrow ADDRESS.RUN
```

SECTION 8
SUMMARY AND CONCLUSIONS

The conversion of the program to the VAX-11/780 computer has yielded a new version of ADDRESS which is free of the problem size restrictions of the previous CYBER version. Any size problem can be handled using the ADDRESS setup and compilation procedure given in Section 7.2. The new equation solver is now much faster even using the slower VAX system.

Standard sized material dimensions can now be used. The code contains a standard of sizes which can be changed if needed by the user. Optimization results can now be obtained which reflect these standard properties.

The addition of a geometry and contour plotter has shown to be highly useful. Both pre- and postprocessing can be performed using standard plotting software such as PLOT10. Contours of the deflection pattern are useful in determining where deflection constraints need to be applied.

The Sturm sequence frequency check allows the user to determine the number of frequencies in a specified range. Shifts in the frequency spectra can be monitored for various damage conditions.

The ADDRESS code is demonstrated for several example structures. Both truss and wing structures are considered. The VAX version of ADDRESS is shown to yield lower-weight structures in some cases run using the CYBER version.

By design variable linking, the large variations in member sizes near damage can be avoided. The user can specify elements which are constrained to be resized as a macro-element. Damage assessment calculations are also aided by using the severity analysis option. This enables the user to obtain a first order accurate assessment of critical damage location before optimization or detailed analysis runs are performed.

REFERENCES

1. Taylor, R. F., "Automated Design of Damage Resistant Structures," AFWAL-TR-82-3087, Vol. I, Theory and Application, October 1982.

2. Taylor, R. F., "Automated Design of Damage Resistant Structures," AFWAL-TR-82-3087, Vol. II - Program User's Manual, October 1982.

3. Venkayya, V. B. and V. A. Tischler, "OPTSTAT - A Computer Program for the Optimal Design of Structures Subjected to Static Loads," AFFDL-TM-FBR-79-67, June 1979.

4. Venkayya, V. B., N. S. Khot, and F. E. Eastep, "Vulnerability Analysis of Optimized Structures," AIAA Journal, Vol. 16, No. 11, November 1978, pp. 1189-1195.

5. Bathe, K-J and E. L. Wilson, Numerical Methods in Finite Element Analysis. Prentice-Hall, 1976, pp. 478-481.

6. Conte, S. D. and C. deBoor, Elementary Numerical Analysis. McGraw-Hill, 1980, pp. 193-195.

Request for Software (ADDRESS)

1. In order to process a request for the ADDRESS software, fill out the attached "Statement of Terms and Conditions for Release of Air Force Owned or Developed Computer Software Packages" form and return it to AFWAL/FIFSD, Wright-Patterson AFB, OH 45433-6553.

2. The software cannot be sent if you indicate the program will be used on a government contract (Item 2 of the form). In such a case, you must obtain the code as a Government Furnished Program (GFP) from the contracting office in charge of the contract.

3. Please send a magnetic tape to AFWAL/FIFSD. ADDRESS will be recorded on the tape after the software release request is processed. Also, please state any requirements concerning transfer of the code to magnetic tape. The following parameters are normally used with 9 track tapes:

VAX VMS standard copy format (recommended)

OR

1600 BPI tape density
80 record length
EBCDIC character code
20 records per block

STATEMENT OF TERMS AND CONDITIONS RELEASE OF AIR FORCE-OWNED OR DEVELOPED COMPUTER SOFTWARE PACKAGES

Date _____

1. In accordance with the provisions of AFR 300-6, release of the following US Air Force software package (computer programs, systems descriptions, and documentation) is requested:

ADDRESS - Automated Design of Damage Resistant Structures

2 The requested software package will be used for the following purpose:

Such use is projected to accrue benefit to the government as follows:

3. I/we will be responsible for assuring that the software package received will not be used for any purpose other than shown in paragraph 2 above; also, it will not be released to anyone without prior approval of the Air Force. Further, the release of the requested software package will not result in competition with other software packages offered by commercial firms.

4. I/we guarantee that the provided software package, or any modified version thereof, will not be published for profit or in any manner offered for sale to the government; it will not be sold or given to any other activity or firm, without the prior written approval of the Air Force. If this software is modified or enhanced using government funds, the government owns the results, whether the software is the basis of, or incidental to a contract. The government may not pay a second time for this software or the enhanced or modified version thereof. The package may be used in contract with the government but no charge may be made for its use.

5. The US Air Force is neither liable or responsible for maintenance, updating or correcting any errors in the software provided.

6. I/we understand that no material subject to national defense security classification or proprietary rights was intended to be released to us. I/we will report promptly the discovery of any material with such restrictions to the Air Force approving authority. I/we will follow all instructions concerning the use or return of such material in accordance with regulations applying to classified material, and will make no further study, use, or copy such material subject to security or proprietary rights marking.

7. I/we understand that the software package received is intended for domestic use only. It will not be made available to foreign governments nor used in any contract with a foreign government.

Signature of Requestor	Signature of Air Force Approving Authority
Name of Requestor	Name/Title of Air Force Approving Authority
Organization/Address	Organization/Location
City, State and Zip Code	

U.S. Government Printing Office: 1986 748-001/60531

END

12-86

DTIC